SecOVID-Authentifikationssystem
Installation, Konfiguration und Benutzung

Version 4.1.2
April 2010

Inhaltsverzeichnis

1 Produktbeschreibung KOBIL SecOVID **3**
1.1 Einleitung . . . 3
1.2 Wie funktioniert KOBIL SecOVID? . . . 3
1.3 Der Einmalpasswortgenerator für den Benutzer . . . 5
1.3.1 Chipkarte und SecOVID Reader III / KOBIL mIDentity . . . 5
1.3.2 SecOVID Token . . . 6
1.3.3 SecOVID SoftToken . . . 6
1.4 Administration und Personalisierung . . . 6
1.4.1 Administration . . . 6
1.4.2 Personalisierung . . . 9
1.5 Der SecOVID-Server . . . 9
1.5.1 Authentifikation, Autorisierung und Accounting . . . 10
1.5.2 Backup-System und Hochverfügbarkeit (optional) . . . 12
1.6 Anwendungsszenarien . . . 12
1.7 Sicherheit des SecOVID-Systems . . . 14

2 Installation **16**
2.1 SecOVID-Serversystem . . . 20
2.1.1 Installationsvoraussetzungen . . . 20
2.1.2 Installation des SecOVID-Servers . . . 21
2.1.3 Installation der SecOVID-Administrationstools . . . 24
2.1.4 Installation von FreeRADIUS zur Unterstützung weiterer Möglichkeiten von RADIUS . . . 28
2.1.5 Installation des SecOVID-Midlet-Generators zur Erstellung von SoftToken-Applikationen für Mobiltelefone und PDAs . . . 28
2.1.6 Web-Interface zum Ändern der Server-PIN . . . 30
2.1.7 Erster Funktionstest . . . 30
2.2 SecOVID-Backup- und Hochverfügbarkeitssystem* . . . 33
2.2.1 Funktionsweise . . . 35
2.2.2 Installation . . . 38
2.2.3 Konfiguration des Hochverfügbarkeitssystems . . . 41
2.3 Clientseite . . . 46
2.3.1 Installation eines RADIUS-Servers* . . . 46
2.3.2 Installation eines RADIUS-Clients* . . . 47
2.3.3 Installation eines TACACS+-Clients* . . . 47
2.3.4 Installation einer Firewall* . . . 47
2.3.5 Installation: Schutz der Einwahl über Windows 2000/XP/2003 RAS* . . . 47
2.3.6 Installation des Apache-Webservers mit SecOVID-Authentifikationsmodul* . . . 47
2.3.7 Installation des Internet Information Server (IIS) mit SecOVID-Authentifikationsmodul* 47
2.3.8 Installation des Internet Security and Acceleration Server mit SecOVID-Authentifikationsmodul* . . . 48
2.3.9 Installation und Konfiguration: Schutz eines UNIX-Netzes* . . . 48

2.3.10 Installation eines RSA-Clients* . . . 52
2.4 Das GetPWD Programm . . . 52

3 Initiale Konfiguration und Bedienung im laufenden Betrieb 53
3.1 SecOVID-Serversystem . . . 53
3.1.1 Konfiguration des SecOVID-Servers . . . 53
3.1.2 Bedienung des SecOVID-Administrationstools . . . 59
3.1.3 Kommandozeilen-Administrationstool . . . 81
3.1.4 Administrations-Bibliothek . . . 90
3.1.5 Konfigurationsbeispiele für FreeRADIUS um weitere Möglichkeiten von RADIUS zu unterstützen . . . 96
3.1.6 Erstellung der SoftToken-Applikationen für Mobiltelefone und PDAs . . . 97
3.1.7 Distribution der SoftToken-Applikation zu Mobiltelefonen und PDAs . . . 98
3.1.8 Migration von Fremdauthentifikationssystemen auf SecOVID . . . 101
3.2 Clientseite . . . 103
3.2.1 Konfiguration eines RADIUS-Servers* . . . 103
3.2.2 Konfiguration eines RADIUS-Clients* . . . 104
3.2.3 Konfiguration eines TACACS+-Clients* . . . 105
3.2.4 Konfiguration einer Firewall* . . . 105
3.2.5 Konfiguration: Schutz der Einwahl über Windows 2000/XP/2003 RAS* . . . 105
3.2.6 Konfiguration des Apache-Webservers* . . . 106
3.2.7 UNIX-Schutz: Benutzung der Arbeitsplatzrechner* . . . 112
3.2.8 Konfiguration eines RSA-Clients* . . . 112
3.3 Das GetPWD Programm . . . 112
3.3.1 Linux und Solaris . . . 112
3.3.2 Windows . . . 112

4 Deinstallation 114

5 Lizenzvereinbarung 116
5.1 Lizenz . . . 116
5.2 Berechtigungsnachweis . . . 117
5.3 Gebühren und Abgaben . . . 117
5.4 Gewährleistung . . . 117
5.5 Haftungsbeschränkung . . . 117
5.6 Allgemein . . . 118

6 Support 119

Kapitel 1

Produktbeschreibung KOBIL SecOVID

1.1 Einleitung

Das Problem – Unsichere Passwörter In vielen IT- und Internet-Anwendungen ist es nötig, dass die tatsächliche Identität eines Benutzers festgestellt werden kann. Webserver, Internet-Anwendungen und Computernetzwerke müssen vor unberechtigtem Zugriff geschützt werden. So muss zum Beispiel beim Online-Banking ein Kunde beweisen, dass er auf sein Konto zugreifen darf. Außendienstmitarbeiter müssen nachweisen, dass sie berechtigt sind, durch die Firewall oder den Router Ihrer Firma auf ihre Daten zuzugreifen. Geheime Passwörter – immer noch weitgehend zur Identifikation gebraucht – sind zu unsicher. Sie können erraten, weitergegeben oder bei der Übertragung mitgehört und dann missbräuchlich eingesetzt werden. Trojanische Pferde warten als harmlos erscheinende Programme nur darauf, dass der rechtmäßige Benutzer das nächste Mal sein geheimes Passwort eingibt, um dieses anschließend unbemerkt einem Hacker zuzusenden. Damit stellen herkömmliche Passwörter eine bedrohliche Sicherheitsschwachstelle ansonsten guter Netzwerkprodukte und Serversysteme dar. Häufig sollen die zur Überprüfung der Identität der Benutzer (Authentifikation) notwendigen Daten sowie deren Berechtigungsprofile (Autorisierung: „Was darf der Benutzer überhaupt auf meinem System tun?“) zentral auf einem Server gespeichert werden.

Unsere Lösung – KOBIL SecOVID SecOVID-Einmalpasswörter (auch One-Time-Passwort, OTP, genannt) bieten eine wesentlich höhere Sicherheit bei der Authentifikation als normale Passwörter. Auf den Computern der Benutzer muss in der Regel keinerlei Soft- oder Hardware installiert werden, so dass diese vollkommen mobil bleiben. Teil der Lösung ist der SecOVID-Server, der als zentraler Authentifikations- und Autorisierungsserver gleichzeitig seine Dienste für viele verschiedene Anwendungen und Systeme (z.B. Firewalls, Router, VPN-Gateways, Webserver) verrichten kann.

1.2 Wie funktioniert KOBIL SecOVID?

Jeder Benutzer des Systems wird wahlweise mit dem SecOVID Token, SecOVID SoftToken oder einer persönlichen SecOVID-Chipkarte mit dem SecOVID Reader III bzw. KOBIL mIDentity ausgestattet. Will sich ein Benutzer bei einem System anmelden, so muss er zum Nachweis seiner Identität das für ihn augenblicklich gültige (Einmal-)Passwort eingeben. Dieses kennt zunächst selbst der Benutzer nicht, es wird vom SecOVID Token, dem SecOVID SoftToken bzw. von der SecOVID-Chipkarte bei Bedarf erzeugt. Die SecOVID Chipkarte bzw. das SecOVID SoftToken ist mit einer individuellen PIN geschützt, die der Benutzer vor der Erzeugung des Einmalpassworts eingeben muss. Ist die PIN korrekt, wird das Einmalpasswort berechnet und angezeigt bzw. im Falle von KOBIL mIDentity direkt an die Anwendung übergeben.
Beim SecOVID Token wird das Einmalpasswort auf Knopfdruck erzeugt und (optional zusammen mit der ServerPIN) zur Anmeldung eingegeben.

SecOVID Token

- Benutzer-individueller geheimer Schlüssel fest vorprogrammiert im Token, wird auf Diskette zum Import in den Server mitgeliefert
- OTP-Erzeugung auf Knopfdruck
- Anzeige des OTPs am Display
- Diebstahlschutz / Zwei-Faktor-Authentisierung durch optionale ServerPIN
- Eingabe des OTPs durch den Benutzer in die Anmeldemaske der Anwendung, ggf. mit ServerPIN

SecOVID SoftToken

- Benutzer-individueller geheimer Schlüssel wird bei Erstellung des SoftTokens generiert
- OTP-Erzeugung nach PIN Eingabe am mobilen Endgerät (Zwei-Faktor-Authentisierung)
- Anzeige des OTPs am Display des mobilen Endgeräts
- Eingabe des OTPs durch den Benutzer in die Anmeldemaske der Anwendung

SecOVID Reader III mit SecOVID Chipkarte

- Benutzer-individueller geheimer Schlüssel wird bei Personalisierung der Chipkarte generiert
- Chipkarten können zentral oder dezentral massenpersonalisiert werden
- OTP-Erzeugung nach Eingabe der Chipkarten-PIN am SecOVID Reader III
- Offline Modus: Anzeige des OTPs am Display des SecOVID Reader III und manuelle Eingabe in die Anmeldemaske der Anwendung
- Online Modus: Übertragung des OTPs an den PC mit der GetPWD Anwendung
- Die Chipkarte kann auch mit weiteren Anwendungen genutzt werden, z.B. PKI

KOBIL mIDentity mit SecOVID Chipkarte im SIM Format

- Benutzer-individueller geheimer Schlüssel wird bei Personalisierung der Chipkarte generiert
- Chipkarten können zentral oder dezentral massenpersonalisiert werden
- OTP-Erzeugung nach Eingabe der Chipkarten-PIN
- Automatisches Einfügen des OTP in die Anmeldemaske der Anwendung (Simple Sign On)
- Die Chipkarte kann auch mit weiteren Anwendungen genutzt werden, z.B. PKI

Jedes Passwort wird nur ein einziges Mal vom System akzeptiert (Einmalpasswort), so dass der Benutzer beim nächsten Anmeldevorgang erneut eine Anfrage an sein Token bzw. seine Chipkarte zu starten hat. Außenstehende (Hacker) können nicht in das System einbrechen, da sie nicht über ein Token oder eine Chipkarte mit in der SecOVID-Datenbank registrierten geheimen Informationen verfügen und gültige Passwörter ohne ein solches Token oder eine Chipkarte in der Praxis nicht vorhergesagt werden können. Zudem kennen Hacker nicht die geheime PIN bzw. die ServerPIN, mit der die Chipkarten bzw. die SecOVID Token der Benutzer geschützt sind. Bei SecOVID basiert die Authentifikation demnach auf zwei Faktoren *(Zwei-Faktor-Authentifikation)*:

1. Etwas, was der Benutzer *besitzt* (Token, SoftToken oder Chipkarte)
2. Etwas, was der Benutzer *weiß* (die geheime PIN).

Komponenten eines funktionierenden Gesamtsystems mit SecOVID

Auf Benutzerseite

- Wahlweise pro Benutzer: ein SecOVID Token, SoftToken mit Handy/PDA oder Chipkarte mit SecOVID Reader III oder KOBIL mIDentity
- Keine spezielle Software. Der Benutzer verwendet die Software, die er bisher benutzt hat (bzw. GetPWD Anwendung / KOBIL mIDentity Software für mehr Komfort)

Auf Serverseite

- Die SecOVID-Server-Software für Authentifikation, Autorisierung und Accounting
- Die SecOVID-Administrationstools zum Verwalten von Benutzern sowie deren Tokens und Chipkarten
- Je nach Anwendung:
  - keine weitere Software, wenn SecOVID in ein Produkt mit RADIUS-Interface integriert werden soll. (Bsp.: Firewalls, Router, VPNs verfügen meistens über ein RADIUS-Interface.)
  - ein SecOVID-Authentifikationsmodul, welches in die zu schützende Anwendung integriert werden muss, falls kein RADIUS-Interface vorhanden ist (siehe unten)

1.3 Der Einmalpasswortgenerator für den Benutzer

Einmalpasswörter können wahlweise mit folgender Hardware generiert werden:

- SecOVID-Chipkarte und SecOVID Reader III bzw. KOBIL mIDentity
- SecOVID Token
- SecOVID SoftToken in Verbindung mit einem Handy oder PDA

Auf der SecOVID-Chipkarte, im SecOVID Token und im Handy/PDA ist eine Applikation „Einmalpasswortgenerator“ enthalten. Dieser speichert das zuletzt generierte Einmalpasswort (anfangs ein initiales Einmalpasswort) und den geheimen, nicht auslesbaren Triple-DES-Schlüssel. Das nächste Einmalpasswort wird im Wesentlichen durch Triple-DES-Verschlüsselung des letzten Einmalpasswortes berechnet. Die Triple-DES-Verschlüsselung wird durch die SecOVID-Chipkarte, das SecOVID Token oder die Software des SoftTokens durchgeführt.

1.3.1 Chipkarte und SecOVID Reader III / KOBIL mIDentity

1.3.1.1 Die SecOVID-Chipkarte

Es wird eine Reihe von zertifizierten kryptographischen Chipkarten unterstützt, derzeit Siemens CardOS M4.01a, M4.2 und M4.3 sowie TCOS 2.0. Die Kombination der SecOVID-Anwendung mit weiteren Applikationen auf der gleichen Karte ist möglich.
Eine Portierung des SecOVID-Einmalpasswortgenerators auf Chipkarten anderer Hersteller ist möglich, sofern diese Chipkarte die technischen Voraussetzungen dazu erfüllt, beispielsweise den Triple-DES-Verschlüsselungsalgorithmus zur Verfügung stellt.

1.3.1.2 SecOVID Reader III

Zum Generieren von Einmalpasswörtern wird der Taschenkartenleser standardmäßig *nicht* an den Computer angeschlossen. Er arbeitet off-line und wird per Batterie betrieben. Optional kann der SecOVID Reader III auch an den Computer angeschlossen werden, dann arbeitet er als vollwertiges Chipkartenterminal der Klasse 2. Die Karten-PIN kann ausspähsicher über die Tastatur des Readers eingegeben werden und wandert niemals in den Computer. Strom bezieht das Chipkartenterminal dann über den Computer.

1.3.1.3 KOBIL mIDentity

Die KOBIL mIDentity Plattform bietet vielfältige Sicherheits-Mechanismen, so z.B. den smartcard basierten Logon am Netzwerk oder am Arbeitsplatz-PC und den sicheren Datensafe. Als Sicherheitsanker dient immerzu die SecOVID Chipkarte, mit der auch bei mIDentity die Einmalpasswörter erzeugt werden können. Als Komfortfunktion wird das sogenannte SSimple Sign Onängeboten, womit die Einmalpasswörter direkt in die Anmeldemaske der Anwendung eingefügt werden, so dass in diesem Fall kein manuelles Abtippen notwendig ist.

1.3.2 SecOVID Token

Das SecOVID Token ist ein kompaktes Gerät, welches dem Benutzer auf Knopfdruck ein OTP erzeugt. Es stellt dieses auf dem Display dar. Der Anwender hat weiter nichts zu tun, als das im Display angezeigte Einmalpasswort über die Tastatur seines PCs oder Notebooks einzugeben, ggf. zusammen mit der optionalen Server-PIN, die als Diebstahlschutz dient und vom SecOVID Server geprüft wird.
Technische Daten:

- 1 Taste zur OTP-Erzeugung
- Display: 1x8 nummerisch und Grafik-Symbole
- Stromversorgung: Knopfzelle CR 2025

1.3.3 SecOVID SoftToken

Das SecOVID SoftToken ermöglicht es Einmalpasswörter (one-time-passwords, OTP) softwarebasierend auf mobilen Endgeräten zu generieren. Zu solchen Geräten zählen Mobiltelefone (Voraussetzung: WAP/WTLS, java-fähig) sowie marktübliche PDAs (ab Palm OS 3.5 / Pocket PC 2002). Diese Lösung stellt eine Alternative zu den bisherigen hardwarebasierenden Geräten, wie dem SecOVID Reader/SecOVID Reader Plus oder dem SecOVID-Token, dar und bietet dem Benutzer des SecOVID-Systems größtmögliche Flexibilität.
Nach der Installation der SofToken-Applikation auf dem jeweiligen mobilen Endgerät, ist es möglich durch Eingabe einer vierstelligen PIN Einmalpassworte zu generieren und diese auf dem Display des Gerätes darzustellen. Die einzugebende PIN kann zudem jederzeit durch den Benutzer geändert werden. Dazu erfolgt eine Abfrage der alten PIN zur Verifikation des Benutzers und es muss zweimal die neue PIN eingegeben werden. Nach erfolgreicher Verifikation der alten PIN wird die neue PIN aktiviert.
Des Weiteren ist es wie bei dem SecOVID Token möglich die Einmalpassworte mit einer Server-PIN zu kombinieren.

1.4 Administration und Personalisierung

1.4.1 Administration

1.4.1.1 Administrationstools

Administrationstool mit GUI Zur Verwaltung der SecOVID-Benutzer, Tokens, SoftTokens und SecOVID-Chipkarten kann das SecOVID-Administrationstool mit GUI (graphischer Oberfläche) eingesetzt werden. Dieses ist für die Betriebssysteme Windows, Linux und Solaris verfügbar. Die Administration kann lokal am Server oder remote erfolgen. Die Kommunikation zwischen Administrationstools und dem SecOVID-Server wird verschlüsselt.
Das Administrationstool mit GUI bietet ASCII-, LDAP- und ADS-Schnittstellen zum Import sowie eine ASCII-Schnittstelle zum Export von Userdaten.
Die wichtigsten Eigenschaften:

- Graphische Sicht auf SecOVID-Benutzer und SecOVID-Administratoren, welche in der SecOVID-Datenbank gespeichert sind. Angezeigt werden:
  - Benutzername (login)
  - Name
  - Vorname
  - Gruppenzugehörigkeit
  - Für Benutzer: Einmalpasswort-Generatortyp
  - Für Administratoren: Rolle/ Rechte des Administrators (Helpdesk, erweiterter Helpdesk oder voller Admin)
  - Für Benutzer: Status (freigeschaltet, gesperrt, Anzahl Fehlversuche in Folge)

  Zusätzlich können für jeden Benutzer in einem zusätzlichen Informationsfenster folgende Informationen angezeigt werden:
  - Welche zusätzlichen Generatoren kann der Benutzer zur Authentifikation benutzen?
  - Wann wurde der Generator personalisiert?
  - Wann hat sich der Benutzer zuletzt erfolgreich authentifiziert?
  - Bis wann ist der Benutzer ggf. temporär gesperrt?
  - Welche Seriennummer hat das Token bzw. die Chipkarte des Benutzers. Beachten Sie, dass die Kartennummer nur für Chipkarten gespeichert wird, die mit den Tools von SecOVID-Version 4.0 oder später erzeugt werden, für zuvor personalisierte Chipkarten kann also keine Seriennummer angezeigt werden.
- Management von SecOVID-Benutzern und SecOVID-Administratoren sowie deren SecOVID-Chipkarten, Tokens und SoftTokens
- lauffähig unter SUN Solaris, Linux und Windows

Weitere Features:

a) Aktionen mit Benutzern/ Administratoren
   - Neues Anlegen einzelner Benutzer oder Administratoren (optional: Unter Berücksichtigung der spezifischen Policy für die Bildung von Benutzernamen)
   - Neues Anlegen einer großen Zahl anonymer Benutzer (optional: Unter Berücksichtigung der spezifischen Policy für die Bildung von Benutzernamen)(nachträgliche Änderung des anonymen Usernamens möglich),
   - Sperren, Entsperren
   - Löschen
   - Ändern bereits vorhandener Einträge
   - Import/ Export (ASCII-Schnittstelle; LDAP-und ADS-Schnittstelle; Verknüpfung von Chipkarten- und Tokendaten mit Einträgen aus fremden Datenquellen (LDAP, ADS, ACE/Server, ASCII-Tabelle)
   - Drucken eines PIN-Briefes
   - Ändern der Server-PIN eines Token-Users (für den Benutzer über ein separates Webinterface möglich)

b) Aktionen mit Chipkarten (für SecOVD-Administratoren und SecOVID-Benutzer)
   - Erzeugen/ Personalisieren (bis zu zehn Passwort-Generatoren auf einer Chipkarte)
   - Löschen eines Passwort-Generators, Löschen aller Generatoren
   - Freischalten einer gesperrten Chipkarte mittels PUK

- Resynchronisierung einer Chipkarte (oder eines Tokens)

c) Gruppenverwaltung
Zuweisung einer oder mehrerer Gruppen zu Benutzern und Administratoren

d) Sortierfunktionen
Alphabetisches Sortieren nach verschiedenen Keys wie Loginname, Nachname etc.

e) Suchfunktionen
Suchen nach verschiedenen Kriterien

f) Konfigurierbarkeit

- Erreichbarkeit der SecOVID-Datenbank
- Länge von PIN und PUK auf den zu personalisierenden SecOVID-Chipkarten
- Erreichbarkeit des Chipkartenlesers (COM-Port) zum Personalisieren von Chipkarten

g) Remote-Administrierbarkeit
Mit starker chipkartenbasierter Authentifikation des Administrators (Challenge-Response) und anschließender Verschlüsselung aller zu übertragenden Daten (Triple-DES, 168 Bit).

Kommandozeilen-Administrationstool Alternativ zum graphischen Administrationstool kann ein rein textbasiertes Administrationstool (Kommandozeilentool) zur Verwaltung von SecOVID-Benutzern und Chipkarten eingesetzt werden. Dieses weist fast die volle Funktionalität des graphischen Tools (GUI) auf und ist ebenfalls für die Betriebssysteme Windows, Linux und Solaris verfügbar. Das textbasierte Tool eignet sich bestens zur Integration der SecOVID-Administration in bereits vorhandene kundenspezifische Administrationstools.
Syntax des Kommandozeilen-Administrationstools:
`SecAdm [ Optionen [ Parameterliste ] ]`
Erläuterung:
Das Programm lässt sich von der Kommandozeile aus mit Optionen und Parametern starten, welche die gewünschte Aktion festlegen. Das Programm liefert bei jeder Aktion Rückgabewerte zurück, die den Erfolg oder Misserfolg einer Aktion anzeigen. Standardinput und Standardoutput können durch die entsprechenden Operatoren umgeleitet werden.
Beispiel:
`SecAdm -new_user login=string1 [lastname=string2] [firstname=string3][pin=string4]` legt einen neuen Benutzer mit den angegebenen Daten in der SecOVID-Datenbank an. Dabei ist die Angabe des Parameters `login` notwendig. Die anderen Parameter sind optional. Wenn keine `pin` angegeben ist, wird automatisch eine zufällige PIN generiert.
Das Kommandozeilen-Administrationstool kann sowohl zur lokalen als auch zur Remote-Administrierbarkeit des SecOVID-Servers eingesetzt werden. Es werden die gleichen Sicherheitsmechanismen wie im Administrationstool mit GUI verwendet (siehe oben). Das Kommandozeilentool lässt sich sogar im „Remote-remote-Modus“ betreiben: Der Administrator greift remote über eine eigene (Web-)Applikation auf das Kommandozeilentool zu, welches remote den Server administriert.

Administrations-Bibliothek Schließlich steht der gesamte Umfang der Administrationstools auch in Form einer Bibliothek zur Integration in eigene Verwaltungsprogramme zur Verfügung. Auch hier finden die gleichen Sicherheitsmechanismen wie für die Administrationstools Verwendung.

1.4.1.2 Datenbank

Die erforderlichen Authentifikationsdaten der SecOVID-Benutzer werden in einer für die SecOVID-Anwendung Performance-optimierten internen Datenbank gespeichert. Benutzerprofile (Berechtigungslisten) können RADIUS- und TACACS+-konform in entsprechenden Dateien gespeichert werden.

1.4.2 Personalisierung

1.4.2.1 Personalisierung von Chipkarten

Der Betreiber des SecOVID-Systems erwirbt leere Chipkarten, d.h. Chipkarten ohne jegliche Anwendungsdaten, und schreibt selbst mit Hilfe der SecOVID-Administrationstools die Applikation „Einmalpasswortgenerator“ auf die Chipkarte.

Zur Personalisierung, also zum initialen Schreiben des Einmalpasswortgenerators auf die Chipkarte, stehen folgende Möglichkeiten zur Verfügung:

a) Manuelles Einlegen jeder einzelnen Chipkarte
Die Benutzer, für die ein Einmalpasswortgenerator auf eine Chipkarte geschrieben werden soll, werden selektiert (im SecOVID-Administrationstool mit GUI per Mausklick, im Kommandozeilen-Administrationstool per Angabe der Benutzernamen). Anschließend müssen die Chipkarten einzeln in das am Administrationsrechner angeschlossene Chipkartenterminal eingelegt werden, ehe der Schreibvorgang, der wenige Sekunden in Anspruch nimmt, beginnen kann.

b) Automatisches Beschreiben einer Vielzahl von Chipkarten
Mittels des SecOVID-Administrationstools oder des textbasierten Administrationstools können SecOVID-Benutzerdaten exportiert werden (Verschlüsselung der Daten), so dass die Einmalpasswortgeneratoren anschließend in einem Zug mittels einer Chipkartenkodiermaschine auf Chipkarten geschrieben werden können. Optional kann hierbei zeitgleich der Kartenkörper der Chipkarte bedruckt werden (Name des Inhabers, Foto usw.).

c) Dezentrale Massenpersonalisierung mit Hilfe des mIDentity Manager Professional

1.4.2.2 Personalisierung von mobilen Endgeräten (SoftTokens)

Der Systembetreiber erstellt auch SecOVID SoftTokens selbst und überspielt diese anschließend auf die mobilen Endgeräte. Dazu verwendet er das SecOVID-Administrationstool, um die entsprechenden Benutzer auszuwählen und für diese die geheimen SoftToken-Datensätze zu erzeugen, welche die Grundlage zur Applikationserstellung bilden. Im nächsten Schritt kommt der SecOVID-Midlet-Generator zum Einsatz, welcher dazu dient, aus den geheimen Datensätzen die zugehörigen SoftToken-Applikationen zu generieren, welche dann auf dem jeweiligen Endgerät installiert werden. Die Installation erfolgt für Mobiltelefone durch einen Download per WAP, bzw. für PDAs unter dem Betriebssystem Palm OS 3.5 oder neuer, per https-Download.
PDAs mit den Betriebssystemen Pocket PC bzw. Windows CE stellen insofern eine Ausnahme dar, als es für diese Geräte eine Anwendung gibt, die für alle Benutzer identisch ist. Aus diesem Grund muss man hier keine Midlets generieren und somit auch nicht den SecOVID-Midlet-Generator verwenden. Die Personalisierung erfolgt durch das Kopieren des durch das Administrationstool exportierten geheimen Datensatzes auf das jeweilige Zielgerät. Näheres entnehmen Sie bitte den Kapiteln 2.1.5 und 3.1.6.

1.4.2.3 Personalisierung von SecOVID Tokens

SecOVID Tokens werden hingegen bereits bei der Produktion in sicherer Umgebung personalisiert (programmiert). Der Kunde erhält zusammen mit den SecOVID Tokens eine Diskette mit den dazugehörigen Datensätzen zum Import in den SecOVID Server. Der Kunde ordnet den SecOVID Tokens nachträglich (mit dem SecOVID-Administrationstool) die Benutzernamen zu.

1.5 Der SecOVID-Server

Der SecOVID-Server ist die Serversoftware, die Passwörter (im allgemeinen Einmalpasswörter) überprüft *(Authentifikation)*, Berechtigungslisten des Benutzers speichert *(Autorisierung)* und die Benutzer-Zugriffe protokolliert *(Accounting)*.

Der SecOVID-Server ist dazu in der Lage, weil er Zugriff auf die SecOVID-Datenbank hat, in der für jeden Benutzer die gleichen (geheimen) Informationen gespeichert sind, die sich auf der Chipkarte des betreffenden Benutzers befinden. So kann der Server softwareseitig ebenfalls das nächste für einen Benutzer gültige Einmalpasswort berechnen und somit entscheiden, ob ein vorgelegtes Passwort korrekt ist oder nicht. Der SecOVID-Server akzeptiert auch Einmalpasswörter, die seiner Meinung nach für den Benutzer nicht aktuell sind, sondern in der zukünftigen Folge gültiger Einmalpasswörter erst an fünfter oder sechster Stelle erscheinen (die Passwortsuchtiefe ist konfigurierbar). Dadurch ist das System auch robust gegen Verlust von Passwörtern bei der Übertragung oder Fehlbedienung durch Benutzer.

1.5.1 Authentifikation, Autorisierung und Accounting

Der SecOVID-Server ist ein Serverprozess, der ständig auf Passwortanfragen lauscht. Dabei kann er drei Arten von Passwortanfragen handhaben, d.h. er verfügt über drei Schnittstellen: eine RADIUS-Schnittstelle, eine TACACS+-Schnittstelle und eine RSA-Schnittstelle.
RADIUS (Remote Authentication Dial-In User Service) ist ein de facto-Standard für eine zentralisierte Benutzerauthentifikation und -autorisierung in großen Rechnernetzen. An der Benutzerauthentifikation und -autorisierung sind verschiedene Rechner beteiligt. Diese sind in folgender Client-Server-Architektur angeordnet: In einem Netz (Internet) gibt es viele verschiedene Rechner (Network Access Servers (NAS) oder RADIUS-Clients), an denen Benutzer einen Remote-Dienst in Anspruch nehmen wollen und sich dazu authentifizieren müssen. Diese RADIUS-Clients führen die Benutzerauthentifikation nicht selbst durch, sondern delegieren diese Aufgabe an einen für das Netz zentralen Rechner, den RADIUS-Server. Die wesentlichen Bestandteile der Kommunikation zwischen RADIUS-Server und RADIUS-Client sind verschlüsselt (mit der Ausnahme der übertragenen Benutzernamen).
Der SecOVID-Server ist ein RADIUS-Server (RADIUS Version 2), der jedoch Einmalpasswortanfragen prüfen kann und über weitere Schnittstellen (TACACS+- und RSA-Schnittstelle) verfügt. Wie die meisten RADIUS-Server kann auch der SecOVID-Server Passwortanfragen an einen weiteren RADIUS-Server weiterleiten und somit für ausgewählte Benutzer als Proxy-Server dienen. Der SecOVID-Server kann wie jeder RADIUS-Server statische Passwörter überprüfen, Benutzerprofile speichern und alle möglichen RADIUS-Attribute handhaben.
TACACS+ (Terminal Access Controller Access Control System) ist ein von Cisco entwickelter de facto-Standard für eine zentralisierte Benutzerauthentifikation und -autorisierung in großen Rechnernetzen. TACACS+ folgt den gleichen Prinzipien wie RADIUS. Der SecOVID-Server kann als TACACS+-Server mit erweiterter Funktionalität angesehen werden.
Der SecOVID-Server kann Passwortanfragen über ein drittes proprietäres Interface, das RSA-Interface, entgegen nehmen. Bei Benutzung dieser RSA-Kommunikation verschlüsselt der SecOVID-Client seine Passwortanfrage mit dem öffentlichen Schlüssel des SecOVID-Servers. Nur der SecOVID-Server kann die Anfrage mit seinem privaten Schlüssel entschlüsseln. Seine Antwort an den SecOVID-Client signiert der SecOVID-Server mit seinem privaten Schlüssel. Das Verwenden des RSA-Protokolls ist vorteilhaft, wenn der SecOVID-Client nicht in geschützter Umgebung steht, wie beispielsweise ein UNIX-Rechner in einem großen Computerarbeitsraum. Das RADIUS-Protokoll hätte hier genau wie das TACACS+-Protokoll den Nachteil, dass der zum Entschlüsseln notwendige Schlüssel (Shared Secret) evtl. auf dem SecOVID-Client ausgespäht werden könnte.
Der SecOVID-Server protokolliert jede Passwort-Anfrage: Unter welchem Benutzernamen wurde wann von welchem SecOVID-Client (z.B. RADIUS-Client) aus eine Passwortanfrage an den SecOVID-Server gerichtet? Was war das Resultat der Passwortüberprüfung?
Die wesentlichen Features des SecOVID-Servers auf einen Blick:

Authentifikation (RADIUS- und TACACS+-konform)

- Alle Authentifikationen werden von einem einzigen Security-Server, dem SecOVID-Server, durchgeführt.
- Einfachste Integration des SecOVID-Servers in existierende IT-Infrastrukturen auf Grund der Kommunikationsschnittstellen des SecOVID-Servers:
  - RADIUS

  - TACACS+
  - (- RSA, 1024 Bit)
- Unterstützte Authentifikationsverfahren:
  - statisches Passwort (PAP, CHAP, MSCHAP)
  - Einmalpasswort (one-time password, OTP) oder `<Server-PIN><OTP>` (PAP, CHAP, MSCHAP)
  - Challenge-Response, basierend auf Triple-DES, mit bis zu 8 Bytes langen Challenges und Responses (nur für SecOVID-Administratoren)
- Überprüfung weiterer Credentials wie IP-Adresse des Benutzers möglich
- Temporäre Festlegung auf ein Authentifikationsverfahren pro Benutzer
- Schutz gegen Brute Force Attack (Durchprobieren aller möglichen Passwörter)
- Suchtiefe für Einmalpasswörter konfigurierbar
- Unterstützt Resynchronisierung

Autorisierung

- Verwaltung von Benutzerprofilen (Berechtigungslisten): RADIUS- und TACACS+-konform

Accounting

- Protokollierung aller Benutzerzugriffe (proprietär bzw. TACACS+-konform)

PROXY

- Einsatzfähig auch als Proxy-Server für andere SecOVID-Server, RADIUS-Server oder ACE/ Server
- Multithreading-fähig, maximale Anzahl paralleler Threads konfigurierbar
- Migration von Benutzern von Fremdauthentifikationsprodukten (z.B. RSA Security ACE/ Server) auf SecOVID mit minimalem administrativem Aufwand

Lastenverteilung

- Statische Lastenverteilung durch Nutzung der Proxy-Funktionalität

Abbildung von Organisationsstrukturen

- Benutzern und Administratoren werden SecOVID-Benutzergruppen zugewiesen, die Abteilungen, Niederlassungen oder Mandanten eines Unternehmens entsprechen *(Mandantenfähigkeit von SecOVID)*. SecOVID-Administratoren können nur die Benutzer der eigenen Gruppe(n) administrieren.
- SecOVID-Administratoren können zudem verschiedene Rollen zugewiesen werden:
  - Helpdesk (rein lesender Zugriff)
  - erweiterter Helpdesk (lesender Zugriff, Benutzer sperren und entsperren)
  - Voller Administrator (Vollzugriff)

Unterstützte Plattformen

- PC Intel i86/ SuSE Linux bzw. openSUSE 8.3 (oder neuer)
- PC Intel i86/ Windows 2003 Server
- PC Intel i86/ Windows 2000 ab Service Pack 3
- PC Intel i86/ Windows XP ab Service Pack 1
- PC Intel i86/ Windows Vista
- Sun Sparc/ Solaris 9 (oder neuer)

1.5.2 Backup-System und Hochverfügbarkeit (optional)

Werden die Benutzeranmeldungen in den verschiedensten Anwendungen über einen einzigen Authentifikationsserver abgewickelt, so ist die ständige Verfügbarkeit dieses Authentifikationsservers extrem wichtig. Der SecOVID-Server erweist sich in der Praxis als extrem stabil. Es könnte aber einen physikalischen Schaden an der Netzwerkverbindung zum SecOVID-Server oder an der Hardware (z.B. Festplatte) des SecOVID-Servers geben, wodurch der SecOVID-Server ausfiele. Das SecOVID-System verfügt über ein ausgeklügeltes Backup- und Hochverfügbarkeitssystem, das optional in Betrieb genommen werden kann und die dauerhafte Verfügbarkeit eines SecOVID-Servers mit aktuellen Benutzerdaten gewährleistet.

Backup-System Durch eine Lösung auf Applikationsebene können Sie die SecOVID-Benutzerdaten auf mehreren Rechnern redundant halten (Backup-System). Werden administrative Änderungen auf dem SecOVID-Hauptserver (bzw. dessen Datenbank) vorgenommen, dann werden automatisch die gleichen Änderungen auf den SecOVID-Backupservern durchgeführt. Zudem sendet der SecOVID-Hauptserver stets eine Kopie der erhaltenen Passwortanfragen an seinen Backupserver. Jede Veränderung der Benutzerdaten wird somit automatisch auf den Backup-Rechnern nachgezogen, so dass die SecOVID-Backupserver immer über aktuelle Benutzerdaten verfügen.

Hochverfügbarkeit Im Falle eines Ausfalls des SecOVID-Hauptservers könnten Sie manuell einen der Backup-Server zum Hauptserver machen, oder Sie können automatisch Hochverfügbarkeit des SecOVID-Systems erzielen, indem der Backup-Server beim Ausfall des Hauptservers automatisch an dessen Stelle einspringt. Im letzteren Falle haben Sie vielfältige Konfigurationsmöglichkeiten. Beispielsweise kann der Backupserver bei entsprechender Konfiguration beim Ausfall des Hauptservers dessen IP-Adresse übernehmen, so dass die anfragenden SecOVID-Clients (z.B. Firewall) nur eine einzige IP-Adresse für den Authentifikationsserver kennen. Alternativ könnte der Backup-Server beim Ausfall des Hauptservers unter einer anderen IP-Adresse erreichbar sein. In diesem Falle ist dafür zu sorgen, dass den anfragenden SecOVID-Clients diese IP-Adresse bekannt ist.

1.6 Anwendungsszenarien

SecOVID kann immer dann mit geringem Integrationsaufwand eingesetzt werden, wenn bisher zur Benutzerauthentifikation normale Passwörter (d.h. Passwörter, die selten oder nie geändert werden) eingesetzt wurden. Viele Software- und Hardwareprodukte (z.B. moderne Firewalls und Router) verfügen bereits über eine RADIUS-Schnittstelle. In diesem Falle arbeiten solche Produkte nach geeigneter Konfiguration auf Anhieb so wie gerade beschrieben mit dem SecOVID-System zusammen. Andere Produkte, die in der Standardversion noch nicht über eine RADIUS-Schnittstelle verfügen (z.B. einige Webserver) bieten häufig die Möglichkeit, ein spezielles RADIUS-Authentifikationsmodul nachzuinstallieren und arbeiten anschließend in gewünschter Weise als NAS mit dem SecOVID-Server zusammen. Zudem existieren Produkte, denen ein von KOBIL entwickeltes RSA-Authentifikationsmodul nachinstalliert werden kann, so dass diese Produkte über die RSA-Schnittstelle mit dem SecOVID-Server kommunizieren können.

Das SecOVID-System eignet sich hervorragend für folgende Einsatzbereiche. In Klammern ist jeweils die verwendete Kommunikationsschnittstelle zum SecOVID-Server angegeben:

Sicheres lokales Login in Ihrem Firmennetzwerk SecOVID verbessert die schwache Authentifikation der heutigen (Netzwerk-)Betriebssysteme:

- Citrix MetaFrame/Presentation Server (RADIUS)
- Linux ((open)SuSE Version 8.3 oder neuer) (eigenes PAM-Modul, RSA)
- UNIX (Solaris 9 oder neuer) (eigenes PAM-Modul, RSA)

Sicheres Remote Login für Ihre Filialen, Außendienstmitarbeiter und Kunden (Telearbeit)

- *Remote Access über das Internet:*
  Wenn Sie Benutzern die Einwahl in Ihr Firmennetz über das Internet ermöglichen wollen, so werden Sie typischerweise eine moderne Firewall einsetzen, welche solche Benutzer authentifiziert.
  Mit SecOVID-Einmalpasswörtern können Sie die bislang schwache Authentifikation entscheidend stärken und somit unberechtigte Benutzer von Ihrem Netzwerk fernhalten. Dabei haben weder Ihre Benutzer noch Sie in Ihrem Firmennetzwerk spezielle Software zu installieren.

  In der Praxis haben sich u.a. folgende Firewalls als vollkommen problemlos im Zusammenspiel mit dem SecOVID-System erwiesen:

  - Check Point Firewall-1 (RADIUS)
  - Symantec/ Axent Raptor (RADIUS)
  - Watchguard (RADIUS)
  - Cisco PIX (RADIUS)

- *Einwahl über das Telefonnetz:*
  Wenn Sie Benutzern die Einwahl in Ihr Firmennetz über das Telefonnetz ermöglichen wollen, so werden Sie eine heute gängige Lösung auswählen. Beispielsweise können Sie unter Windows den RAS (Remote Access Service) einrichten. Oder Sie installieren einen modernen Router, der die Telefonanrufe der Benutzer entgegennimmt, die sich in Ihr Firmennetzwerk einloggen möchten. In beiden Fällen kann SecOVID die bislang schwache Authentifikation entscheidend verbessern. Dabei haben weder Sie in Ihrem Firmennetzwerk noch Ihre Benutzer weitere spezielle Hard- oder Software zu installieren. Ihre Benutzer sind somit vollkommen mobil. (Natürlich haben Sie einmalig den SecOVID-Server in Ihrem Firmennetzwerk zu installieren.)
  In der Praxis haben sich u.a. folgende Router oder Einwahlsysteme als vollkommen problemlos im Zusammenspiel mit dem SecOVID-System erwiesen:

  - Windows 2000/XP/2003 RAS (RADIUS)
  - Cisco-Router (RADIUS und TACACS+)
  - Ascend-Router (RADIUS), Bsp.: Ascend MAX 4000
  - Bintec-Router (RADIUS)

- *Kopplung mit einem Virtual Private Network (VPN)*
  Wenn Sie möchten, dass Ihre Benutzer nach der Einwahl gesichert mit Ihrem Firmennetz kommuniziert (Verschlüsselung der übertragenen Daten und Überprüfung der Datenintegrität), so installieren Sie zusätzlich zu Firewall, Router oder Einwahlserver (siehe oben) ein VPN. SecOVID arbeitet mit allen gängigen VPNs zusammen.

  In der Praxis haben sich u.a. folgende VPNs als vollkommen problemlos im Zusammenspiel mit SecOVID erwiesen:

  - Check Point VPN-1
  - Stonesoft StoneGate
  - NCP VPN Gateway (RADIUS)
  - F-Secure VPN+
  - Cisco VPN Gateway
  - Conware VPN Gateway

Starke Authentifikation für Ihr e-Business

- *Online-Bereitstellung von Informationen oder Internet-Applikationen für einen ausgewählten Personenkreis*
  Durch die Anbindung von Webservern (z.B. Apache-Server) oder des MS ISA-Servers an SecOVID kann der Zugriff auf einzelne Verzeichnisse des Webservers dem SecOVID-Schutz unterstellt werden. Somit können Sie beispielsweise dafür sorgen, dass nur registrierte Benutzer (Ihre Mitarbeiter oder Ihre Kunden) auf sicherheitssensitive oder kostenpflichtige Informationen auf dem Webserver zugreifen können. Auf die gleiche Weise kann der Zugriff auf Internet-Applikationen wie Outlook Web Access mit Einmalpasswörtern geschützt werden. Durch die Kopplung mit SSL (Secure Socket Layer) können Daten nach erfolgter Authentifikation des Benutzers verschlüsselt übertragen werden. Optional können ausgewählte Benutzer mit Techniken der Public-Key-Kryptographie authentifiziert werden, sofern die Benutzer über ein gültiges Zertifikat verfügen.

- *Absicherung Ihres Internetbanking-Systems*

- *Absicherung Ihres Callcenter-Dienstes (z.B. Telefonbanking)*

Starke Authentifikation in Ihrer eigenen Applikation
Die Integration des SecOVID-Einmalpasswortmechanismus in Ihre eigenen Applikationen können Sie innerhalb kürzester Zeit mit dem *SecOVID-Toolkit* vornehmen. Dieses finden Sie auf der SecOVID CD-ROM unter `/Development`. Das SecOVID-Toolkit besteht aus Headern, Shared Objects und Dokumentation. Die Shared Objects beinhalten RADIUS-Clients, die Sie aus Ihrer eigenen Applikation mit einer entsprechenden Parameterliste (z.B. IP-Adresse und Port des SecOVID-Servers) aufrufen können. Ein solcher RADIUS-Client führt die komplette RADIUS-konforme Kommunikation mit dem SecOVID-Server aus, d.h. er kodiert die Passwort-Anfrage im entsprechenden Format, verschlüsselt sie gemäß RADIUS-Standard und sendet sie an den SecOVID-Server. Schließlich liefert Ihnen der RADIUS-Client einen Return-Code zurück, welchen er vom SecOVID-Server erhalten hat. Dieser Return-Code kodiert die Antwort des SecOVID-Servers (1 = accept, 0 = reject, -1, ..., -11 = spezifischer Fehler). Es wird ebenfalls erläutert, wie proprietäre Java-Anwendungen an das SecOVID-System angebunden werden.

Starke Authentifikation für SAP R/3
Durch die Anbindung von SAP R/3 können sich Benutzer mit SecOVID-Einmalpasswörtern über die SAP-eigene Benutzeroberfläche (SAPGUI) an SAP R/3 anmelden[1]. Alle zwischen SAPGUI und SAP R/3-Server ausgetauschten Daten werden verschlüsselt (Triple-DES, 168 Bit). Benutzer können wahlweise über Einmalpasswörter oder unter Verwendung zertifikatsbasierter Public-Key-Technologie (Produkt KOBIL Smart Key) authentifiziert werden. Bei Verwendung mehrerer SAP R/3-Server ist nur ein einmaliges Logon notwendig (Single Sign-On).

1.7 Sicherheit des SecOVID-Systems

SecOVID-Einmalpasswörter bieten eine viel stärkere Authentifikation als normale Passwörter. Somit schließt SecOVID die größte Sicherheitslücke in heutigen Internet- oder Netzwerkapplikationen. Die folgenden Angriffe bringen einem Angreifer auf SecOVID in der Praxis keinen Erfolg bzw. werden unterbunden. Entsprechende Angriffe auf Systeme, die auf normale Passwörter setzen, verlaufen hingegen in der Regel erfolgreich und kompromittieren somit die Sicherheit des Systems:

- Eine Replay-Attacke, d.h. das Wiedereinspielen eines ehemals gültigen Einmalpasswortes.
  Denn Einmalpasswörter werden nur ein einziges Mal akzeptiert.

- Das Erraten eines gültigen Einmalpasswortes.
  Denn es gibt sehr viele (100 Millionen) verschiedene Möglichkeiten für das jedes Mal scheinbar zufällig gewählte Einmalpasswort.

[1] Das entsprechende SecOVID-Authentifikationsmodul KOBIL eSecure ist auf Anfrage erhältlich.

- Eine Brute-Force-Attacke, d.h. das Durchprobieren aller möglichen Passwörter für einen Benutzernamen.
  Denn werden für den gleichen Benutzernamen zehn ungültige Passwörter in Folge eingegeben, so wird der betreffende Benutzer dauerhaft gesperrt. Des Weiteren wird nach jeder Falscheingabe ein Timeout hochgezählt, der den Benutzer temporär sperrt. Das heißt nach der ersten Falscheingabe muss der Benutzer 4 Sekunden warten bis er das nächste OTP angeben kann, nach der zweiten Falscheingabe 8 Sekunden, dann 16 Sekunden, dann 32 Sekunden usw. Dadurch werden Denial of Service Attacken abgewehrt, die darauf abzielen, Benutzerkonten zu sperren.

- Der Versuch, ein gültiges Einmalpasswort aus mitgehörten gültigen Einmalpasswörtern zu berechnen.
  Denn man kann mathematisch beweisen, dass die Vorausberechnung eines künftig gültigen Passwortes durch einen Angreifer nur dann möglich wäre, wenn das symmetrische Verschlüsselungsverfahren Triple-DES mit einer Schlüssellänge von 168 Bit gebrochen wäre. Genau dies wird seit mehr als zwanzig Jahren weltweit von den Fachleuten der Kryptologie vergeblich versucht und erscheint daher höchst unwahrscheinlich, selbst unter Einsatz enormer Rechenpower.

- Das Stehlen des Taschenkartenlesers SecOVID Reader III und der SecOVID-Chipkarte oder des SecOVID Tokens eines Benutzers.
  Denn im Taschenkartenleser SecOVID Reader ist keine sicherheitssensitive Information gespeichert. Zur Benutzung der SecOVID-Chipkarte muss die richtige individuelle Karten-PIN eingegeben werden. Nach dreimaliger fehlerhafter Eingabe der PIN sperrt sich die Chipkarte automatisch. Das „Clonen“ der Chipkarte und das Auslesen des geheimen Triple-DES-Schlüssels von der Chipkarte ist technisch nicht möglich.

- Das Stehlen des Handys/PDAs eines Benutzers.
  Denn selbst wenn man an die Daten der aufgespielten Sicherheits-Software gelangt, ist es nicht möglich ohne Kenntniss der korrekten PIN an die darin verschlüsselten Geheimnisse zu kommen (Beachten Sie jedoch, das in diesem Falle eine Brute-Force-Attacke auf diese PIN mglich ist).

- Das Ausspähen der einmaligen Datenübertragung zum Handy/PDA.
  Denn zur Datenübertragung werden sichere Verbindungen wie SSL für den Download zum PDA bzw. WTLS zum OTA-Transfer auf das Handy benutzt. (OTA - Over The Air)

- Das Ausspähen des SecOVID-Servers.
  Denn der SecOVID-Server wird in einer physikalisch abgesicherten Umgebung installiert und geeignet konfiguriert.

Die Sicherheit besteht nicht darin, das Verfahren zur Generierung eines Einmalpasswortes geheimzuhalten.

Kapitel 2

Installation

Um die von Ihnen gewünschte Anwendung mit dem SecOVID-System abzusichern, müssen Sie einige Arbeitsschritte durchführen. Die folgende Tabelle gibt Ihnen Aufschluss darüber. In der zweiten Spalte wird der Arbeitsschritt benannt. Die dritte Spalte enthält das für den jeweiligen Arbeitsschritt aufzurufende Installationsprogramm auf Ihrer CD. `$UNIX` bezeichnet dabei das Betriebssystem, auf dem Sie den SecOVID-Server installieren möchten (`linux` oder `solaris`). Die vierte Spalte verweist auf die Abschnitte des Handbuches, in denen Sie detailliertere Informationen finden. Bitte halten Sie bei der Installation die von uns empfohlene Reihenfolge ein.

Achten Sie darauf, dass die von Ihnen verwendeten Softwarekomponenten Bestandteil der gleichen SecOVID-Version sind. Insbesondere ist der SecOVID-Server ab Version 1.10 nicht kompatibel mit früheren Versionen der RSA-Clienten; SecOVID-Server bzw. Admintool ab Version 2.0 sind nicht kompatibel mit früheren Versionen von Server bzw. Admintool.

Installation unter Linux, Solaris

	Arbeitsschritt	**aufzurufendes Installationsskript / Programm**	**Details siehe Abschnitt(e)**
1	Installation SecOVID-Server	`$UNIX/install.sh`	2.1.2, 3.1.1
2	Installation SecOVID-Administrationstools	`$UNIX/install_admintool.sh`	2.1.3, 3.1.2
3	Starten des SecOVID-Servers (Lizenzdatei notwendig!)	`/etc/SecOvid/secovid.sh`	3.1.1
4	Starten des Administrationstools (ggf. Nachkonfigurieren)	`/etc/SecOvid/SecOvid_admin.sh`	3.1.2
5	Test von Administrationstool und SecOVID-Server (Benutzer mit Karte anlegen oder Token-Daten importieren, Passwortanfragen an SecOVID-Server stellen (mit `radping`, `tacping` oder `ovid_ping`), siehe 2.1.3 und 2.1.7)	`/etc/SecOvid/radping -u username/password -S SecovidServerIP[:serverPort] -K secretKey` bzw. `/etc/SecOvid/tacping -u username/password -S SecovidServerIP -K secretKey` bzw. `/etc/SecOvid/ovid_ping -u username/password`	3.1.1
6	Installation Ihrer zu schützenden Anwendung	-im Falle eines RADIUS-Clients oder eines TACACS+-Clients: siehe Handbuch zu Ihrer Anwendung -Apache-WWW-Server: `$UNIX/apache_install.sh` -andere Anwendungen: siehe 2.3, 3.2	2.3, 3.2

	Arbeitsschritt	**aufzurufendes Installationsskript/ Programm**	**Details siehe Abschnitt(e)**
7	Für RADIUS-Clients: Eintragen des Rechners der zu schützenden Anwendung im SecOVID-Server (Eintragen der IP-Adresse des Clients und des shared secret)	Editieren Sie `/etc/SecOvid/clients`	3.1.1
8	Konfiguration der zu schützenden Anwendung. Für RADIUS-Clients und TACACS+-Clients: siehe Handbuch des betreffenden Produktes, i.a. Eintragen der SecOVID-Server-IP-Adresse und des shared secret. Für RSA-Clients: Eintragen von IP-Adresse, Portnummer und Public Key des SecOVID-Servers	Für RADIUS-Clients: Häufig ist auf dem Client `/etc/raddb/clients` zu editieren. Für RSA-Clients unter Windows: Editieren Sie `\WinNT\system32\drivers\etc\ovid_config` und kopieren Sie die vom SecOVID-Server erzeugte Datei `ovid_comm.pub` vom Server als `\WinNT\system32\drivers\etc\ovid_comm.pub` auf den Client	3.2.1, 3.2.4, 3.2.5, 3.2.2, 3.2.6 3.2.7

Installation unter Windows

	Arbeitsschritt	**Installationsprogramm / Programm**	**Details siehe Abschnitt(e)**
1	Installation SecOVID-Server	`d:\win32\server\SECOVID Server.exe`	2.1.2, 3.1.1
2	Installation SecOVID-Administrationstools	`d:\win32\admin\SECOVID_admintool.exe`	2.1.3, 3.1.2
3	Starten des SecOVID-Servers (Lizenzdatei notwendig!)	Systemsteuerung > Verwaltung > Dienste > SecOVID Server	3.1.1
4	Starten des Administrationstools	Programme > KOBIL Systems > SecOVID admintools > WxOVID	3.1.2
5	Test von Administrationstool und SecOVID-Server (Benutzer mit Karte anlegen oder Token-Daten importieren, Passwortanfragen an SecOVID-Server stellen (mit `radping`, `tacping` oder `ovid_ping`), siehe 2.1.3 und 2.1.7)	`c:\etc\SecOvid\radping -u username/password -S SecovidServerIP[:serverPort] -K secretKey` bzw. `c:\etc\SecOvid\tacping -u username/password -S SecovidServerIP -K secretKey` bzw. `c:\etc\SecOvid\ovid_ping -u username/password`	3.1.1
6	Installation Ihrer zu schützenden Anwendung	-im Falle eines RADIUS-Clients oder eines TACACS+-Clients: siehe Handbuch zu Ihrer Anwendung -andere Anwendungen: siehe 2.3, 3.2	2.3, 3.2
7	Für RADIUS-Clients: Eintragen des Rechners der zu schützenden Anwendung im SecOVID-Server (Eintragen der IP-Adresse des Clients und des shared secret)	Editieren Sie `c:\etc\SecOvid\clients`	3.1.1
8	Konfiguration der zu schützenden Anwendung. Für RADIUS-Clients und TACACS+-Clients: siehe Handbuch des betreffenden Produktes, i.a. Eintragen der SecOVID-Server-IP-Adresse und des shared secret. Für RSA-Clients: Eintragen von IP-Adresse, Portnummer und Public Key des SecOVID-Servers	Für RSA-Clients unter Windows: Editieren Sie `c:\WinNT\system32\drivers\etc\ovid_config` und kopieren Sie die vom SecOVID-Server erzeugte Datei `ovid_comm.pub` vom Server als `c:\WinNT\system32\drivers\etc\ovid_comm.pub` auf den Client	3.2.1, 3.2.4, 3.2.5, 3.2.2, 3.2.6 3.2.7

2.1 SecOVID-Serversystem

Im vorliegenden Handbuch bezeichnen wir mit dem SecOVID-Serversystem die Gesamtheit aller Dateien, die zum SecOVID-Server (inkl. SecOVID-Datenbank) und zu den SecOVID-Administrationstools gehören.

2.1.1 Installationsvoraussetzungen

Voraussetzungen für die Installation und Inbetriebnahme des SecOVID-Servers:

- Eine der folgenden Plattformen:
  - PC i86/ SuSE Linux bzw. openSUSE 8.3 oder neuer
  - PC i86/ Windows 2000 (ab SP3), XP (ab SP1), 2003 Server oder Vista
  - SUN SPARC/ Solaris 9 oder neuer
- CD-ROM-Laufwerk
- freier USB-Port zum Anschluss des Chipkartenterminals
- 32 MB RAM, zusätzlich 660 Bytes pro Benutzer (z.B. 1 MB für 1.500 Benutzer, 10 MB für 15.000 Benutzer)
- 20 MB freier Plattenplatz, zusätzlich mind. 660 Bytes pro Benutzer (800 Bytes empfohlen) (z.B. 1 MB für 1.500 Benutzer, 10 MB für 15.000 Benutzer)
- funktionsfähige TCP/IP-Netzwerkanbindung

Voraussetzungen für die Installation und Inbetriebnahme der SecOVID-Administrationstools (GUI-Version und Kommandozeilentool):

- Eine der folgenden Plattformen:
  - PC i86/ SuSE Linux bzw. openSUSE 8.3 oder neuer
  - PC i86/ Windows 2000 (ab SP3), XP (ab SP1), 2004 Server oder Vista
  - SUN SPARC/ Solaris 9 oder neuer
- CD-ROM-Laufwerk
- freier USB-Port zum Anschluss des Chipkartenterminals
- 32 MB RAM
- 15 MB freier Plattenplatz
- funktionsfähige TCP/IP-Netzwerkanbindung
- Für die GUI-Version des SecOVID-Administrationstools unter UNIX-Systemen zusätzlich: Die graphische Oberfläche X muss laufen.

Im folgenden bezeichnet `$UNIX` eines der Verzeichnisse `/linux` oder `/solaris`, je nachdem, unter welchem der genannten Betriebssysteme Sie den SecOVID-Server (bzw. die SecOVID-Administrationstools) installieren möchten.

2.1.2 Installation des SecOVID-Servers

2.1.2.1 Standard-Installation unter UNIX-Systemen

Wenn Sie eine bereits vorhandene Installation updaten wollen, erstellen Sie zunächst ein Backup Ihrer Daten und gehen dann genauso wie im folgenden beschrieben vor.
Zur Standard-Installation des SecOVID-Servers (empfohlen!) gehen Sie wie folgt vor:

1. Loggen Sie sich als `root` ein.
2. Falls dies nicht bereits automatisch geschehen ist, so mounten Sie Ihr CD-ROM-Laufwerk, so dass Programme auf der CD ausführbar sind: `mount -o exec /cdrom`
3. Starten Sie das Installationsskript `$UNIX/install.sh` auf der CD, lesen Sie die Ausgaben des Installationsskriptes zu Ihrer Information und folgen Sie den Anweisungen und Empfehlungen des Installationsskriptes. Wählen Sie insbesondere ein Installationsverzeichnis aus. Im Falle eines Updates können Sie entweder ein neues Verzeichnis wählen und nach der Installation Ihre Konfigurationsdateien von Hand aus dem Verzeichnis der alten Version in der Verzeichnis mit der neuen Version kopieren, oder Sie können das Verzeichnis der vorhandenen Installation angeben. Im zweiten Fall wird nur die aktuellen Versionen der Programme installiert, ihre Konfigurationsdateien bleiben unverändert erhalten - sie sollten allerdings beachten, dass Sie in diesem Fall den Server zunächst stoppen sollten, um sicherzustellen, dass die Datei nicht mehr vom Betriebssystem blockiert ist. Zur Installation ist es notwendig, den Kommunikationsinterfaces (z.B. RADIUS, TACACS+) des SecOVID-Servers Portnummern zuzuweisen. Falls Ihre Systemdatei `/etc/services` bereits Einträge für `radius` oder `tacacs` enthält, schlägt Ihnen das Installationsskript ggf. vor, die vorhandenen Einträge zu ändern. Wenn Sie den Empfehlungen des Installationsskriptes folgen, haben Sie in der Regel anschließend einen geringeren Aufwand bei der Konfiguration des SecOVID-Systems.

   Was geschieht beim Ausführen des Installationsskriptes `install.sh`?

   (a) Entpacken des SecOVID-Servers in das gewählte Installationsverzeichnis. Falls noch keine Konfigurationsdateien existieren (wenn es also kein Update ist), werden zusätzlich Default-Konfigurationsdateien entpackt.
   Im Verzeichnis wurde nun der SecOVID-Server installiert. Hier existiert jetzt das Programm `secovid`, welches den SecOVID-Dämon (SecOVID-Serverprozess, Dienst) darstellt, der in Ausführung an vier Ports lauscht, um Anfragen entgegenzunehmen und abzuarbeiten.

   (b) Ergänzungen in der Datei `/etc/services`
   Hier wurden die verschiedenen Interfaces des SecOVID-Dämons zu Portnummern zugeordnet. An dieser Stelle sollten Sie sicherstellen, dass die von SecOVID benutzten Portnummern von keinem anderen Dämon verwendet werden. Die Datei `/etc/services` wurde um folgende Zeilen ergänzt (falls entsprechende Einträge nicht bereits vorher existierten bzw. Sie eine Modifikation abgelehnt hatten):

   ```
   tacacs        49/tcp
   tacacs        49/udp
   radius        1812/udp
   secovid       1647/udp
   secoviddb     1113/udp
   ```

   Der SecOVID-Dämon wartet nun an dem Port namens `tacacs` (im vorliegenden Fall also unter Portnummer `49`) auf TACACS+-konforme Passwortanfragen. Entsprechend wartet der SecOVID-Server am Port namens `radius` (im vorliegenden Fall also unter Portnummer `1812`) auf RADIUS-konforme Passwortanfragen und am `secovid`-Port (Portnummer `1647`) auf RSA-verschlüsselte Passwortanfragen. Das Administrationstool greift über den `secoviddb`-Port (Port `1113`) auf die im Arbeitsspeicher residierende Datenbasis zu.

(c) Erstellen des Startupdateien `/etc/rc.d/rc2.d/S65secovid`
Bem.: Das Default-Init-Verzeichnis unter Solaris lautet `/etc/rc2.d/`.
Genau wie alle Programme im Verzeichnis `/etc/rc.d/rc2.d/` wird auch das Skript `S65secovid` automatisch beim Booten des Rechners ausgeführt. Dies führt zum Starten des SecOVID-Dämons. Von Hand starten Sie den SecOVID-Dämon durch Eingabe von `$(INSTALL_DIR)/secovid.sh` oder von `/etc/rc.d/rc2.d/S65secovid start`.

4. Im folgenden nehmen wir an, dass der SecOVID-Server in `$(INSTALL_DIR)` installiert ist. Versuchen Sie nun den SecOVID-Server zu starten: `$(INSTALL_DIR)/secovid.sh`
Der SecOVID-Server überprüft beim Starten jedes Mal die Gültigkeit der installierten Lizenzdatei `$(INSTALL_DIR)/ovid_licence`. Das Ergebnis dieser Überprüfung finden Sie in der SecOVID-Log-Datei: Das Kommando
`cat $(INSTALL_DIR)/ovid.proto` könnte folgendes ausgeben:
```
Licence for ovid (radius): KOBIL-Demo-Lizenz
Users: 20
valid till: Fri Nov 14 11:00:19 2003
```
Die installierte Lizenzdatei berechtigt in diesem Beispiel zum Betrieb des SecOVID-Servers für bis zu 20 Benutzer bis zum 14.11.2003.
Der letzte Eintrag in der SecOVID-Log-Datei könnte aber auch folgender Gestalt sein:
```
Licence for ovid (radius): KOBIL-Test-Lizenz
Users: 20
valid till: Mon Aug 5 14:55:55 2002
Your licence expired. It was valid till Mon Aug 5 14:55:55 2002.
```
Die installierte Lizenzdatei ist in diesem Beispiel zeitlich nicht mehr gültig. Besorgen Sie sich über Ihren SecOVID-Händler eine gültige Lizenzdatei und installieren Sie diese (siehe unten).

5. Falls Sie eine gültige Lizenzdatei installieren möchten, gehen Sie wie folgt vor:

   (a) Kopieren Sie die gültige Lizenzdatei `ovid_licence` ins Installationsverzeichnis:
   `cp $UNIX/ovid_licence $(INSTALL_DIR)`

   (b) Starten Sie den SecOVID-Server erneut, beispielsweise durch
   `/etc/rc.d/rc2.d/S65secovid restart`

   (c) Überzeugen Sie sich davon, dass der SecOVID-Server läuft, z.B. durch Eingabe von
   `ps aux |fgrep secovid`

6. Die Installation Ihres SecOVID-Servers ist abgeschlossen. Fahren Sie mit der Installation des SecOVID-Administrationstools (siehe Abschnitt 2.1.3) fort.

Bemerkung:
Folgende Dateien werden vom SecOVID-Server beim ersten Start im Verzeichnis `$(INSTALL_DIR)` erzeugt:

- `ovidcomm.key`. Der private Schlüssel des SecOVID-Servers für die gesicherte Kommunikation mit den RSA-Clients. Dieser wird vom SecOVID-Server für die Entschlüsselung von Passwortanfragen verwendet. Der SecOVID-Server erzeugt das Paar aus privatem und öffentlichem Schlüssel bei seinem ersten Start und legt die betreffenden Dateien an. Somit findet die Schlüsselerzeugung bei Ihnen vor Ort statt.

- `ovidcomm.pub`. Der öffentliche Schlüssel des SecOVID-Servers für die gesicherte Kommunikation mit den RSA-Clients. Dieser sollte hier als Backup liegen und muss auf allen RSA-Clientrechnern im vorgesehenen Pfad gespeichert sein.

2.1.2.2 Individuelle Installation unter UNIX-Systemen

Wünschen Sie eine Installation mit besonderen Einstellungen, so kopieren Sie die relevanten Dateien der CD (zumindest `install.sh` und `secovid*.ezp`) auf Ihre Festplatte, modifizieren `install.sh` nach Ihren Wünschen und starten `install.sh` von der Festplatte.
Beachten Sie auch die Hinweise aus dem vorangegangenen Abschnitt.

2.1.2.3 Installation unter Win32-Systemen

1. Loggen Sie sich als `Administrator` ein.

2. Starten Sie das Installationsprogramm `win32\server\SECOVID Server.exe` auf der CD, lesen Sie die Ausgaben des Setups zu Ihrer Information und folgen Sie den Anweisungen und Empfehlungen der Installationsroutine. Zur Installation ist es notwendig, den Kommunikationsinterfaces (z.B. RADIUS, TACACS+) des SecOVID-Servers Portnummern zuzuweisen. Die Installationsroutine überprüft Ihre Systemdatei `\winnt\system32\drivers\etc\services` nach bereits vorhandenen Einträgen für `radius` oder `tacacs`. Sind diese Einträge nicht vorhanden, werden diese automatisch ergänzt.

   Was geschieht beim Ausführen des Installationsskriptes?

   (a) Im gewählten Installationsverzeichnis wird der SecOVID-Server installiert. Hier existiert jetzt das Programm `secovid.exe`, welches den SecOVID-Dämon (SecOVID-Serverprozess, Dienst) darstellt, der in Ausführung an vier Ports lauscht, um Anfragen entgegenzunehmen und abzuarbeiten.

   (b) Ergänzungen in der Datei `c:\WINNT\system32\drivers\etc\services`
   Hier wurden die verschiedenen Interfaces des SecOVID-Dämons zu Portnummern zugeordnet. An dieser Stelle sollten Sie sicherstellen, dass die von SecOVID benutzten Portnummern von keinem anderen Dämon verwendet werden - insbesondere der Internet-Authentifikations-Dienst kann mit dem SecOVID Server um den RADIUS-Port konkurrieren, so dass Sie diesen Service abschalten müssen, um SecOVID benutzen zu können. Dies tut das Setup-Programm automatisch, andere Dienste benötigen allerdings möglicherweise einen manuellen Eingriff. Die Datei `/etc/services` wurde um folgende Zeilen ergänzt (falls entsprechende Einträge nicht bereits vorher existierten bzw. Sie eine Modifikation abgelehnt hatten):

   ```
   tacacs        49/tcp
   tacacs        49/udp
   radius        1812/udp
   secovid       1647/udp
   secoviddb     1113/udp
   ```

   Der SecOVID-Dämon wartet nun an dem Port namens `tacacs` (im vorliegenden Fall also unter Portnummer `49`) auf TACACS+-konforme Passwortanfragen. Entsprechend wartet der SecOVID-Server am Port namens `radius` (im vorliegenden Fall also unter Portnummer `1812`) auf RADIUS-konforme Passwortanfragen und am `secovid`-Port (Portnummer `1647`) auf RSA-verschlüsselte Passwortanfragen. Das Administrationstool greift über den `secoviddb`-Port (Port `1113`) auf die im Arbeitsspeicher residierende Datenbasis zu.

3. Starten Sie den SecOVID-Server: Systemsteuerung > Verwaltung > Dienste > SecOVID Server
   Der SecOVID-Server überprüft beim Starten jedes Mal die Gültigkeit der installierten Lizenzdatei `\etc\SecOvid\ovid_licence`. Das Ergebnis dieser Überprüfung finden Sie in der SecOVID-Log-Datei: Das Kommando
   `type \etc\SecOvid\ovid.proto` könnte folgendes ausgeben:
   ```
   Licence for ovid (radius): KOBIL-Demo-Lizenz
   Users: 20
   valid till: Fri Nov 14 11:00:19 2003
   ```
   Die installierte Lizenzdatei berechtigt in diesem Beispiel zum Betrieb des SecOVID-Servers für bis zu 20 Benutzer bis zum 14.11.2003.
   Der letzte Eintrag in der SecOVID-Log-Datei könnte aber auch folgender Gestalt sein:
   ```
   Licence for ovid (radius): KOBIL-Test-Lizenz
   Users: 20
   valid till: Mon Aug 5 14:55:55 2002
   Your licence expired. It was valid till Mon Aug 5 14:55:55 2002.
   ```
   Die installierte Lizenzdatei ist in diesem Beispiel zeitlich nicht mehr gültig. Besorgen Sie sich über Ihren SecOVID-Händler eine gültige Lizenzdatei und installieren Sie diese (siehe unten).

4. Falls Sie eine gültige Lizenzdatei installieren möchten, gehen Sie wie folgt vor:
   (a) Kopieren Sie die gültige Lizenzdatei `ovid_licence` ins Installationsverzeichnis
   (b) Starten Sie den SecOVID-Server erneut
   Systemsteuerung > Verwaltung > Dienste > SecOVID Server
   (c) Überzeugen Sie sich davon, dass der SecOVID-Server läuft, z.B. durch Eingabe von
   `type ovid.proto`
5. Die Installation Ihres SecOVID-Servers ist abgeschlossen. Fahren Sie mit der Installation des SecOVID-Administrationstools (siehe Abschnitt 2.1.3) fort.
6. Unter Windows kann es notwendig sein, den Rechner neu zu starten, bevor das Admintool eine Verbindung zum SecOVID Server aufnehmen kann.

Bemerkung:
Folgende Dateien werden vom SecOVID-Server beim ersten Start im Installationsverzeichnis erzeugt:

- `ovidcomm.key`. Der private Schlüssel des SecOVID-Servers für die gesicherte Kommunikation mit den RSA-Clients. Dieser wird vom SecOVID-Server für die Entschlüsselung von Passwortanfragen verwendet. Der SecOVID-Server erzeugt das Paar aus privatem und öffentlichem Schlüssel bei seinem ersten Start und legt die betreffenden Dateien an. Somit findet die Schlüsselerzeugung bei Ihnen vor Ort statt.
- `ovidcomm.pub`. Der öffentliche Schlüssel des SecOVID-Servers für die gesicherte Kommunikation mit den RSA-Clients. Dieser sollte hier als Backup liegen und muss auf allen RSA-Clientrechnern im vorgesehenen Pfad gespeichert sein.

2.1.3 Installation der SecOVID-Administrationstools

Die auf der Installations-CD enthaltenen SecOVID-Administrationstools dienen zur Verwaltung von SecOVID-Benutzern, SecOVID Tokens und SecOVID-Chipkarten. Wie bereits in 1 beschrieben, gibt es zwei Varianten von SecOVID-Administrationstools:

- *Administrationstool (GUI).*
  Hier handelt es sich um ein Tool mit graphischer Oberfläche, welches eine komfortable Administration ermöglicht. Die Inbetriebnahme dieses graphischen Tools setzt unter UNIX-Systemen voraus, dass die graphische Oberfläche X läuft. Ggf. müssen Sie also zunächst einen X-Server installieren oder starten!
- *Kommandozeilentool.*
  Hier handelt es sich um ein rein textbasiertes Tool, welches Sie von der Kommandozeile aus starten. Das Kommnadozeilentool bietet alle grundlegenden Funktionen des Administrationstools (GUI). Sinnvolle Einsatzbereiche und die Benutzung des Kommandozeilentools erläutern wir in Abschnitt 3.1.3.

Achtung: Eine **Remote-**Administration (Admintool und SecOVID-Server auf verschiedenen Rechnern) ist nur mit Administrator-Chipkarte möglich. Diese wird benötigt, um eine sichere Kommunikation durchzuführen. Einen ersten „SecOVID-Administratoraccount“ mit Administrator-Chipkarte können Sie nur mit Hilfe eines lokal installierten Administrationstools anlegen. Daher müssen Sie eines der beiden Administrationstools (bisweilen nennen wir sie der Einfachheit halber „Admintools“) in jedem Fall lokal auf dem Rechner des SecOVID-Servers installieren.

Installation unter UNIX-Systemen Sowohl für ein Update einer bestehenden Installation als auch für eine Neuinstallation gehen Sie wie im folgenden beschrieben vor:

1. Falls noch nicht geschehen, mounten Sie die Installations-CD, so dass Programme auf der CD-ROM ausführbar sind.
2. Starten Sie das Skript `$UNIX/install_admintool.sh` von der CD-ROM.

3. Geben Sie zunächst bitte an, in welches Verzeichnis die Administrationstools installiert werden sollen.

4. Danach geben Sie bitte an, in welches Verzeichnis der Treiber für das KOBIL-Chipkartenterminal (shared object) installiert werden soll. Wir empfehlen, das vorgeschlagene Verzeichnis zu akzeptieren. (Sie müssen ein Chipkartenterminal auf Ihrem Rechner installieren, um sich als Administrator an der SecOVID-Datenbank zu authentifizieren oder Einmalpasswortgeneratoren auf eine Chipkarte aufzubringen oder von ihr zu entfernen oder die für eine Remote-Administration erforderlichen Administrator-Chipkarten zu erzeugen.)

5. Geben Sie anschließend bitte an, in welches Verzeichnis andere shared object-Dateien installiert werden sollen, welche das Admintool (GUI) wegen seiner graphischen Oberfläche benötigt. Wir empfehlen erneut, das vorgeschlagene Verzeichnis zu akzeptieren.

Nehmen Sie bitte zu Ihrer Information die Ausgaben des Skriptes zur Kenntnis.
Die eigentlichen Administrationstools wurden im Unterverzeichnis `admin` des gewählten Installationsverzeichnisses installiert, im Installationsverzeichnis selbst finden sich nur Wrapper-Skripte zur Erleichterung des Aufrufs dieser Tools.
Schließen Sie nun Ihr KOBIL-Chipkartenterminal an und befolgen Sie die mitgelieferten Installationshinweise. Mit Hilfe dieses Chipkartenterminals schreibt das Admintool Einmalpasswortgeneratoren auf die Chipkarten, erzeugt Administrator-Chipkarten oder authentifiziert den Administrator an der SecOVID-Datenbank.

Installation unter Win32-Systemen Zur Installation der Administrationstools unter Win32-Systemen gehen Sie wie folgt vor:

1. Starten Sie von der Installations-CD das Setupprogramm für die Treiber Ihres KOBIL-Chipkartenterminals: `/Treiber-Setup/KOBILTreiberSetup.exe`.
   Folgen Sie den Anweisungen des Setupprogramms und schließen Sie das Chipkartenterminal an, wenn Sie dazu aufgefordert werden.

2. Starten Sie von der Installations-CD das Setupprogramm für die Administrationstools:
   `/win32/admin/Setup_admintools.exe`.
   Folgen Sie den weiteren Anweisungen des Setupprogramms.

Starten des SecOVID-Administrationstools (GUI) Beachten Sie, dass das Admintool (GUI) nur Benutzer und Karten verwalten kann, falls der SecOVID-Server in Betrieb ist. Eine Remote-Administration ist nur möglich, wenn Sie bereits über eine am SecOVID-Server registrierte SecOVID-Administratorchipkarte verfügen.

Unter UNIX-Systemen
Achtung: Das Starten des SecOVID-Administrationstools (GUI) erfordert auf UNIX-Systemen die graphische Oberfläche X.
Zur Auswahl der Sprache und des Zeichensatzes wertet das Administrationstool die Umgebungsvariable `LANG` aus. Dabei kann es in Abhängigkeit vom System und dem Wert der Variablen zu leichten Problemen kommen, die sich etwa darin äußern, dass eine deutsche Version des Administrationstools gestartet wird, jedoch Texte, die Umlaute enthalten, nicht richtig oder gar nicht angezeigt werden. Sie können dieses Problem gegebenenfalls umgehen, indem Sie die Umgebungsvariable explizit setzen mittels:

```
LANG=de_DE;export LANG
```

Unter englischem Solaris kann, je nach Version, hier eine Fehlermeldung erfolgen. In diesem Falle können Sie mittels

```
LANG=de;export LANG
```

die Verwendung deutscher Texte erzwingen. Sie starten das SecOVID-Administrationstool (GUI) durch Eingabe von

```
cd /etc/SecOvid
./SecOvid_admin.sh &
```

Falls Sie das Administrationstool in einer englischsprachigen Umgebung starten, erscheint eine harmlose Warnung, dass das Tool keine Datei mit expliziten Übersetzungen der englischsprachigen Originaltexte ins Englische finden kann. Diese Warnung bestätigen Sie gegebenenfalls einfach mit `OK`.
Die Bedienungsanleitung für das Administrationstool (GUI) finden Sie in Abschnitt 3.1.2.
Unter Win32-Systemen
Sie starten das SecOVID-Administrationstool (GUI) über
Start > Programme > KOBIL Systems > SecOVID-Administrationstool.
Falls Sie das Administrationstool in einer englischsprachigen Umgebung starten, erscheint eine harmlose Warnung, dass das Tool keine Datei mit expliziten Übersetzungen der englischsprachigen Originaltexte ins Englische finden kann. Diese Warnung bestätigen Sie gegebenenfalls einfach mit `OK`.
Die Bedienungsanleitung für das Administrationstool (GUI) finden Sie in Abschnitt 3.1.2.

Starten des SecOVID-Kommandozeilentools Beachten Sie, dass das rein textbasierte SecOVID-Kommandozeilentool nur Benutzer und Karten verwalten kann, falls der SecOVID-Server in Betrieb ist. Eine Remote-Administration ist nur möglich, wenn Sie bereits über eine am SecOVID-Server registrierte SecOVID-Administratorchipkarte verfügen.

Die Konfigurationsdatei `Prefs.cfg`
Zudem müssen Sie vor dem Starten des Tools sicherstellen, dass im lokalen Verzeichnis eine Konfigurationsdatei `Prefs.cfg` existiert. Wenn Sie das Administrationstool (GUI) ebenfalls verwenden, wird eine solche Datei automatisch angelegt und beinhaltet die zuletzt von Ihnen verwendeten Einstellungen. Wollen oder können Sie das Administrationstool (GUI) nicht verwenden, so müssen Sie eine solche Datei von Hand anlegen bzw. an Ihre Konfiguration anpassen. Diese Konfigurationsdatei könnte folgenden Inhalt haben:

```
localhost
1113
1
0
0
0
6
6
0
0
```

Die Datei `Prefs.cfg` besteht aus mindestens zehn Zeilen (und höchstens 14 Zeilen), deren Inhalt Sie gemäß Ihrem System Zeile für Zeile anpassen müssen:

Zeile 1: Enthält die IP-Adresse des SecOVID-Servers. In obigem Beispiel ist der SecOVID-Serverprozess also auf dem lokalen Rechner installiert.

Zeile 2: Enthält die Portnummer, an der der SecOVID-Server Administrationsanfragen entgegennimmt. Tragen Sie die Portnummer ein, welche in der Datei `/etc/services` auf dem Rechner des SecOVID-Servers dem Interface `secoviddb` zugeordnet ist. Der Standard-Port für Administrationsanfragen lautet 1113.

Zeile 3: Falls nicht automatisch nach dem COM-Port gesucht werden soll, an dem das KOBIL-Chipkartenterminal angeschlossen ist, so muss hier der korrekte COM-Port eingetragen werden. In obigem Beispiel ist COM-Port 1 angegeben.

Zeile 4: Der Wert `0` bedeutet, dass beim Starten des Kommandozeilentools nicht automatisch nach einem Chipkartenterminal gesucht wird. In diesem Falle muss das Chipkartenterminal an dem in Zeile 3 angegebenen COM-Port angeschlossen sein.

Der Wert `1` bedeutet, dass beim Starten des Kommandozeilentools automatisch nach einem Chipkartenterminal gesucht wird. In diesem Fall wird Zeile 3 ignoriert.
Andere Werte als `0` oder `1` sind nicht zulässig. In obigem Beispiel wird nicht automatisch nach einem Chipkartenterminal gesucht, d.h. wenn der in der Zeile zuvor angegebene Port falsch ist, wird das Tool nicht funktionieren.

Zeile 5: Der Wert `0` bedeutet, dass die (Chipkarten-)PIN bei neu anzulegenden Benutzern rein-numerisch sein soll (nur Ziffern).
Der Wert `1` bedeutet, dass die (Chipkarten-)PIN bei neu anzulegenden Benutzern alphanumerisch sein soll (Buchstaben und Ziffern).
Andere Werte als `0` oder `1` sind nicht zulässig. In obigem Beispiel bestehen künftig zu verwendende (Chipkarten-)PINs nur aus Ziffern.

Zeile 6: Der Wert `0` bedeutet, dass die (Chipkarten-)PUK bei neu anzulegenden Benutzern rein-numerisch sein soll (nur Ziffern).
Der Wert `1` bedeutet, dass die (Chipkarten-)PUK bei neu anzulegenden Benutzern alphanumerisch sein soll (Buchstaben und Ziffern).
Andere Werte als `0` oder `1` sind nicht zulässig. In obigem Beispiel bestehen künftig zu verwendende (Chipkarten-)PUKs nur aus Ziffern.

Zeile 7: Diese Zeile enthält die Länge der (Chipkarten-)PIN für neu anzulegenden Benutzer. Im Beispiel sind (Chipkarten-)PINs für neu anzulegende Benutzer sechsstellig.

Zeile 8: Diese Zeile enthält die Länge der (Chipkarten-)PUK für neu anzulegende Benutzer. Im Beispiel sind (Chipkarten-)PUKs für neu anzulegende Benutzer sechsstellig.

Zeile 9: Es können bis zu 10 verschiedene Einmalpasswortgeneratoren auf eine Chipkarte geschrieben werden (Passwortgenerator Nr. 0 bis Passwortgenerator Nr. 9). Der Wert `1` in Zeile 9 bedeutet, dass beim Schreiben (oder Löschen) eines Einmalpasswortgenerators auf eine Chipkarte immer der in Zeile 10 spezifizierte Default-Generator verwendet werden soll. (Der tatsächlich zu verwendende Passwortgenerator kann in der eigentlichen Anweisung zum Schreiben oder Löschen auf der Chipkarte nochmals angepasst werden.)
Der Wert `0` bedeutet, dass beim Schreiben oder Löschen eines Einmalpasswortgenerators auf der Chipkarte kein Default-Generator benutzt werden soll.
Andere Werte als `0` oder `1` sind nicht zulässig. In obigem Beispiel ist kein Default-Generator gesetzt.

Zeile 10: Diese Zeile enthält den zu verwendenden Einmalpasswortgenerator (Nr. 0, ..., Nr. 9). Dieser Eintrag besitzt nur Relevanz, wenn in Zeile 9 durch den Wert `1` kodiert ist, dass ein Default-Generator verwendet werden soll und wenn in der Anweisung zum Schreiben oder Löschen nicht explizit ein Passwortgenerator angegeben ist. In obigem Beispiel würde Passwortgenerator Nr. 0 als Default-Wert definiert werden, wenn in Zeile 9 die Benutzung eines Default-Wertes aktiviert wäre.

Zeile 11: (optional, d.h. diese Zeile können Sie zunächst leer lassen.) Diese Zeile enthält den Default-Pfad, der beim Exportieren von Benutzereinträgen verwendet wird. Der Pfad zur Export-Datei kann bei jedem Exportvorgang nochmals angepasst werden. Der Default-Eintrag ist `./`, also das aktuelle Verzeichnis.

Zeile 12: (optional, d.h. diese Zeile können Sie zunächst leer lassen.) Diese Zeile enthält den Default-Pfad, der beim Importieren von Benutzereinträgen verwendet wird. Der Pfad zur Import-Datei kann bei jedem Importvorgang nochmals angepasst werden. Der Default-Eintrag ist `./`, also das aktuelle Verzeichnis.

Zeile 13: (optional, d.h. diese Zeile können Sie zunächst leer lassen.) Diese Zeile enthält den Default-Pfad zur Log-Datei, die Probleme beim Importieren und Exportieren von Benutzereinträgen festhält. Der Pfad zur Log-Datei kann bei jedem Importvorgang nochmals angepasst werden. Der Default-Eintrag ist `./OvidLog.log`.

Zeile 14: (optional, d.h. diese Zeile können Sie zunächst leer lassen.) Diese Zeile enthält den Default-SecOVID-Administratornamen. Der Eintrag hat nur Relevanz, wenn Sie im Kommandozeilentool nicht explizit den Administratornamen mit übergeben.

Hinweis: Die Datei `Prefs.cfg` können Sie auch mit dem SecOVID-Administrationstool (GUI) erzeugen. Starten Sie hierzu das SecOVID-Administrationstool (GUI) und nehmen Sie die gewünschten Einstellungen über den Menüpunkt Programm > Konfiguration vor. Die Datei `Prefs.cfg` wird dann im lokalen Verzeichnis auf der Festplatte erzeugt.

2.1.4 Installation von FreeRADIUS zur Unterstützung weiterer Möglichkeiten von RADIUS

Einige Möglichkeiten, die RADIUS bietet, werden von unserer RADIUS-Schnittstelle nicht unterstützt, z.B. Accounting und EAP. Auch bieten andere RADIUS-Implementierungen teilweise die Möglichkeit, das users-File statt als File mittels einer Datenbank zu verwalten, was je nach Anwendungssituation ein Vorteil sein kann.
RADIUS bietet die Möglichkeit, mittels der Proxy-Funktionalität die Vorteile verschiedener Server zu kombinieren, z.B. können Sie einen RADIUS-Server installieren, der alle gewünschten Features enthält und nur die eigentliche Passwort-Verifikation an den SecOVID-Server delegiert, wir diskutieren hier die Details am Beispiel von FreeRADIUS.
Zur Installation benötigen Sie zunächst einen Compiler, under Linux oder Solaris z.B. gcc (http://gcc.gnu.org), unter Windows ist zwingend cygwin erforderlich (http://www.cygwin.com).
Sobald eine geeignete Umgebung zum Compilieren vorhanden ist, können sie FreeRADIUS von `http://www.freeradius.org` beziehen, das KOBIL EAP-MD5-Modul (vgl. Abschnitt 3.1.5) wurde mit FreeRadius in den Versionen 1.0.0 und 1.0.2 getestet.
Das Compilieren und Installieren von FreeRADIUS funktioniert im Prinzip nach dem bekannten Schema von

```
./configure
./make
./make install
```

Allerdings sind einige Details zu beachten:

1. Wenn Sie das EAP-MD5-Modul (vgl. Abschnitt 3.1.5) benutzen wollen, kopieren sie zunächst die Dateien aus dem Verzeichnis `/FreeRadius/rlm_eap_md5` der CD in das Verzeichnis `src/modules/rlm_eap/types/rlm_eap_md5/` der FreeRadius-Distribution.

2. Wenn Sie unter Windows mit cygwin compilieren, ändern Sie vor dem Aufruf von `./configure` in der Datei `src/modules/rlm_eap/Makefile.in` die Zeile
   `RLM_LIBS = -Llibeap -leap`
   in
   `RLM_LIBS =`

3. Wenn Sie unter Verwendung von statischen OpenSSL-Bibliotheken compilieren, rufen Sie statt `./configure` folgenden Befehl auf:

   ```
   LIBS="-L<Pfad_zu_den_Bibliotheken> -lssl -lcrypto" \
   ./configure --with-openssl-includes=<Pfad_zu_den_Headern>
   ```

2.1.5 Installation des SecOVID-Midlet-Generators zur Erstellung von SoftToken-Applikationen für Mobiltelefone und PDAs

Will man das SoftToken System auf mobilen Geräten wie Handys oder Palm-PDAs (ab PalmOS 3.5) einsetzen, so benötigt man spezielle Applikationen für diese Geräte. Diese Applikationen werden Midlets genannt und setzen zur Funktionsfähigkeit eine sogenannte J2ME Runtime-Umgebung voraus (Java 2 Micro Edition). Der SecOVID-Midlet-Generator dient dazu, diese Applikationen zu personalisieren. Die Inbetriebnahme dieses Tools wird in den folgenden Abschnitten erkläutert.

Sollen PDAs mit den Betriebssystemen Windows CE bzw. Pocket PC verwendet werden, so können Sie die nachfolgenden Abschnitte überspringen und direkt zu Kapitel 3.1.7.4 gehen.

Installation unter UNIX-Systemen Zur Installation des SecOVID-Midlet-Generators gehen Sie unter UNIX-Systemen wie folgt vor:

1. Falls noch nicht geschehen, loggen Sie sich als root ein und mounten Sie die Installations-CD, so dass Programme auf der CD-ROM ausführbar sind.
2. Starten Sie das Skript `$UNIX/Softtoken/install_j2me.sh` von der CD-ROM. Beim Ausführen dieses Skriptes wird das J2ME-Entwicklungspaket bestehend aus JDK1.5 (Java Development Kit v1.5), CLDC (Connected Limited Device Configuration v1.0), MIDP (Mobile Information Device Profile v2.0) und Apache Ant installiert.
3. Anschließend entpacken Sie die Datei namens `$UNIX/Softtoken/MidletGenerator.tgz` in ein Verzeichnis Ihrer Wahl (Solaris: Verzeichnis ist bereits entpackt und muss nur an eine beliebige Stelle kopiert werden). In diesem Verzeichnis liegen die benötigten Komponenten zur Midleterstellung.
4. Benutzen Sie das Skript start.sh für einen ersten Funktionstest, nachdem Sie die Installationspfade in der Datei buildconfig.properties angepasst haben. Während dieses Tests wird ein Testtoken verwendet, welches unter `"<IhrPfad>/MidletGenerator/FirstTokens"` zu finden ist. Die PIN zu diesem Testtoken lautet `"123456"`.
5. Für die Erstellung realer Benutzerapplikationen stehen Ihnen zwei Möglichkeiten zur Auswahl (lesen Sie dazu Abschnitt 3.1.6):
   (a) Erstellen der Anwendung für einen einzelnen Benutzer
   (b) Erstellen der Anwendung für mehrere Benutzer

Installation unter Windows-Systemen Zur Installation des SecOVID-Midlet-Generators unter Win32-Systemen gehen Sie wie folgt vor:

1. Falls noch nicht geschehen, loggen Sie sich als Administrator ein.
2. Starten Sie das Setup des Java-SDKs `$WIN32\Softtoken\jdk-1_5_0_06-windows-i586-p.exe`.
3. Entpacken Sie das Verzeichnis `$WIN32\Softtoken\MidletGenerator_Env\apache-ant-1.6.5-bin.zip` unter dem Pfad, unter dem Sie auch das Java-SDK installiert haben (Bsp.: `C:\Programme\java`)
4. Kopieren Sie die Datei `$WIN32\Softtoken\MidletGenerator_Env\ant-contrib.jar` ins lib-Verzeichnis der ANT-Installation (Bsp.: `C:\Programme\Java\apache-ant-1.6.5\lib`)
5. Entpacken Sie das Verzeichnis `$WIN32\Softtoken\MidletGenerator_Env\j2me.zip` unter dem Pfad, unter dem Sie auch das Java-SDK installiert haben (Bsp.: `C:\Programme\java`).
6. Entpacken Sie das Verzeichnis `$WIN32\Softtoken\MidletGenerator.zip` an eine beliebige Stelle (Bsp.: `C:\Programme\KOBIL Systems`)
7. Setzen Sie folgende Umgebungsvariablen:
   (a) JAVA_HOME (Bsp.: `C:\Programme\java\jdk-1_5_0_06`)
   (b) ANT_HOME (Bsp.: `C:\Programme\java\apache-ant-1.6.5`)
   (c) MIDP_HOME (Bsp.: `C:\Programme\java\j2me\midp2.0fcs`)
   (d) PATH (Bsp.: `%PATH%;%JAVA_HOME%\bin;%ANT_HOME%\bin;%MIDP_HOME%\bin`)

8. Benutzen Sie das Skript start.bat für einen ersten Funktionstest, nachdem Sie die Installationspfade in der Datei buildconfig.properties angepasst haben. Während dieses Tests wird ein Testtoken verwendet, welches unter `"<IhrPfad>/MidletGenerator/FirstTokens"` zu finden ist. Die PIN zu diesem Testtoken lautet `"123456"`.
   Für die Erstellung realer Benutzerapplikationen stehen Ihnen zwei Möglichkeiten zur Auswahl (lesen Sie dazu Abschnitt 3.1.6):

   (a) Erstellen der Anwendung für einen einzelnen Benutzer

   (b) Erstellen der Anwendung für mehrere Benutzer

2.1.6 Web-Interface zum Ändern der Server-PIN

Falls Sie SecOVID-Token mit Server-PIN verwenden, können Benutzer über dieses Web-Interface ihre Server-PIN ändern. Wenn Sie noch keinen Apache-Webserver auf Ihrem System installiert haben, folgen Sie den Anweisung „Installation eines neuen Apache-Webservers“. Haben Sie bereits einen Apache-Webserver, lesen Sie die Installationsanweisungen unter Punkt 2.

1. Installation eines neuen Apache-Webservers
   Starten Sie das Installationsskript `$UNIX/apache_install.sh`. Die Module für das Ändern der Server-PIN werden automatisch an die richtige Stelle kopiert. Sie müssen nur noch die IP-Adresse und das „shared secret“ in den Perl-Modulen anpassen. Bitte lesen Sie hierzu die Datei `Readme.txt` im Verzeichnis `$UNIX/apache`.

2. Installation bei einem bereits existierenden Apache-Webserver
   Entpacken Sie das `serverpin_package.tar`-Paket aus dem Verzeichnis `$UNIX/serverpin_web` und kopieren Sie die Daten aus dem cgi-bin-Verzeichnis in das cgi-bin-Verzeichnis, die Daten aus dem htdocs-Verzeichnis in das htdocs-Verzeichnis Ihres Apache-Servers. Abschließend müssen Sie die IP-Adresse und das „shared secret“ in den Perl-Modulen anpassen. Bitte lesen Sie hierzu die Datei `Readme.txt` im Verzeichnis `$UNIX/apache`.

Sie können das Web-Interface unter der folgenden URL nutzen:
`https://<IP-Adresse-Apache-Server>:8443/StartSeite.shtml`

2.1.7 Erster Funktionstest

Sie haben nun den SecOVID-Server sowie die SecOVID-Administrationstools (auf dem Rechner des SecOVID-Servers) installiert. Wir empfehlen, an dieser Stelle einen ersten Funktionstest vorzunehmen. Arbeiten Sie dabei folgende Punkte in der angegebenen Reihenfolge ab:

1. Starten Sie den SecOVID-Server: `/etc/SecOvid/secovid.sh` (Unix) bzw.
   Systemsteuerung > Verwaltung > Dienste > SecOVID Server (Win32)

2. Starten Sie das SecOVID-Administrationstool (GUI) auf dem Rechner des SecOVID-Servers:
   `/etc/SecOvid/SecOvid_admin.sh &` (Unix) bzw.
   Programme > KOBIL Systems > SecOVID admintools > WxOVID (Win32)

3. Überprüfen der Einstellungen.
   Überprüfen Sie unter Programm > Konfiguration die Einstellungen für den SecOVID-Server (Datenbank IP und Datenbank Port) sowie für das Chipkartenterminal. Klicken Sie auf den OK-Button. Jetzt werden alle Einstellungen auf Plausibilität überprüft. Insbesondere wird überprüft,

   - ob am eingestellten Port ein KOBIL-Chipkartenterminal gefunden wird (Dieser Vorgang kann ca. 3 Sekunden in Anspruch nehmen), falls nicht automatisch nach einem Chipkartenterminal gesucht werden soll.

- ob an irgendeinem Port ein KOBIL-Chipkartenterminal gefunden wird (Dieser Vorgang kann pro COM-Port, den ihr Rechner hat, ca. 3 Sekunden dauern), falls die automatische Suche nach einem Chipkartenterminal aktiviert ist.
- ob der SecOVID-Server angesprochen werden kann.

Im positiven Fall werden Ihnen alle im SecOVID-Server registrierten Benutzer angezeigt (u.U. sind noch keine Benutzer registriert).
Im negativen Fall erhalten Sie eine entsprechende Fehlermeldung - Überprüfen Sie die Einstellungen sowie die Installation des Chipkartenterminals. Stellen Sie beim Versuch einer Remote-Administration des SecOVID-Servers zudem sicher, dass Ihre Firewalls und sonstigen Paketfilter die Netzwerk-Kommunikation zwischen SecOVID-Admintool und SecOVID-Server nicht behindern. Das bedeutet, die Firewall muss alle UDP-Pakete durchlassen, die vom SecOVID-Admintool zum SecOVID-Server auf dessen Datenbankport (in der Regel Port 1113, siehe den Eintrag `secoviddb` in der Datei `/etc/services` des SecOVID-Servers) gesendet wird. Zudem muss die Firewall alle UDP-Pakete des SecOVID-Servers an das SecOVID-Admintool (undefinierter Port) durchlassen.

4. Test mit den Benutzern von SecOVID-Chipkarten:
   (a) Legen Sie mit dem Administrationstool einen SecOVID-Benutzer an (siehe Abschnitt 3.1.2).
   (b) Erzeugen Sie für diesen Benutzer eine SecOVID-Chipkarte (siehe Abschnitt 3.1.2).
   (c) Testen Sie mit den im folgenden kurz beschriebenen Testprogrammen, ob
      - gültige Einmalpasswörter des registrierten Benutzers genau einmal akzeptiert, ansonsten abgelehnt werden.
      - falsche Einmalpasswörter von registrierten Benutzern abgelehnt werden.
      - Einmalpasswörter von nicht registrierten oder (vorübergehend) gesperrten Benutzern abgelehnt werden.

5. Test mit den Benutzern von SecOVID Tokens:
   (a) Falls Sie von Ihrem Händler ein SecOVID-Token mit zugehörigem Tokendatensatz auf Datenträger (Diskette) erhalten haben, importieren Sie den entsprechenden Tokendatensatz über den Menüpunkt: Token > Token importieren. (In einer Teststellung liegen die Tokendatensätze unverschlüsselt auf dem Datenträger vor, während Sie nach Abschluss eines Kaufvertrages in der Regel einen Datenträger mit RSA-verschlüsselten (1024 Bit) Tokendatensätzen erhalten. Im letzteren Fall werden Sie beim Importieren der Datensätze dazu aufgefordert, die zum Entschlüsseln der Datensätze geeignete Chipkarte in das Chipkartenterminal einzulegen und die richtige Karten-PIN einzugeben. Diese zum Entschlüsseln notwendige Chipkarte erhalten Sie einmalig von Ihrem Händler. Nur Sie sind in der Lage, die Tokendatensätze zu entschlüsseln. Der entsprechende private Schlüssel ist an keinem weiteren Ort gespeichert.)
   (b) Sie sehen anschließend den neuen Eintrag in der Ansicht. Als Benutzername ist die Seriennummer des Tokens eingetragen. Doppelklicken Sie auf den neuen Eintrag und tragen Sie im Feld Benutzernamen einen Benutzernamen Ihrer Wahl ein. Klicken Sie auf den OK-Button und bestätigen Sie, dass Sie den Eintrag verändern möchten.
   (c) Falls in der Spalte „Generator-Typ“ der Benutzeransicht ein Fragezeichen steht, muss der Generator-Typ angepasst werden. Hierzu müssen Sie unter Programm > Konfiguration auf der Seite Passwort-Generator den Eintrag „*Generator-typ ändern*“ markieren. Dadurch wird der Menüpunkt Datenbank > Generator-Type ändern aktiviert. Markieren Sie alle Einträge, die einen unbekannten Generator-Typ besitzen und ändern diesen unter Datenbank > Generator-Type ändern auf den Generator-Typ Ihrer Token.
   (d) Testen Sie mit den im folgenden kurz beschriebenen Testprogrammen, ob
      - gültige Einmalpasswörter des registrierten Benutzers genau einmal akzeptiert, ansonsten abgelehnt werden.

- falsche Einmalpasswörter von registrierten Benutzern abgelehnt werden.
- Einmalpasswörter von nicht registrierten oder (vorübergehend) gesperrten Benutzern abgelehnt werden.

Ein RADIUS-Testclient: radping Das hier zu verwendende Testprogramm `radping` befindet sich nach der Installation des SecOVID-Servers im Verzeichnis `/etc/SecOvid`. Falls Sie das Testprogramm auf einem anderen Rechner in Ihrem Netzwerk laufen lassen möchten, so kopieren Sie bitte von Ihrer Installations-CD `$UNIX/clients/radping` (im Falle eines UNIX-Zielrechners) oder `\win32\clients\radping\radping.exe` (im Falle eines Windows-Zielrechners) auf die entsprechende Festplatte. Bei `radping` handelt es sich um eine einfache RADIUS-Client-Software, mit der Benutzername und Einmalpasswort testweise an den SecOVID-Server geschickt werden können. Der SecOVID-Server und `radping` müssen zur Verschlüsselung der abzusendenden Daten beide über den gleichen geheimen Schlüssel (secret key) verfügen. Dazu tragen Sie auf dem Rechner des SecOVID-Servers in der Datei `/etc/SecOvid/clients` die IP-Adresse des Hosts von `radping` sowie den secret key ein. Die beiden Informationen sind durch Leerzeichen voneinander getrennt:

```
192.168.1.17 secret
```

Mit `radping` stellen Sie dann folgendermaßen eine Passwortanfrage an den SecOVID-Server:

- `radping -u username/password -S SecovidServerIP[:serverPort] -K secretKey`
- z.B.`radping -u smith/26011735 -S 192.168.1.1 -K secret`

Dabei ist `username` ein im SecOVID-Server registrierter Benutzer und `password` ein für diesen User gültiges Einmalpasswort.
In der Ausgabe von `radping` bedeutet „Acked“ (acknowledged), dass die Kombination Username, Passwort korrekt war, während „Rejected“ die Ablehnung des SecOVID-Servers anzeigt. Die Ausgabe „Timed out“ zeigt Ihnen an, dass die gesicherte Kommunikation zwischen `radping` und dem SecOVID-Server nicht funktioniert hat, beispielsweise weil `radping` und der SecOVID-Server nicht - wie erforderlich - den gleichen secret key verwenden oder weil der SecOVID-Server nicht erreichbar ist.

Ein TACACS+-Testclient: tacping Das hier zu verwendende Testprogramm `tacping` befindet sich nach der Installation des SecOVID-Servers im Verzeichnis `/etc/SecOvid`. Falls Sie das Testprogramm auf einem anderen Rechner in Ihrem Netzwerk laufen lassen möchten, so kopieren Sie bitte von Ihrer Installations-CD `$UNIX/clients/tacping` (im Falle eines UNIX-Zielrechners) oder `\win32\clients\tacping\tacping.exe` (im Falle eines Windows-Zielrechners) auf die entsprechende Festplatte. Bei `tacping` handelt es sich um eine einfache TACACS+-Client-Software, mit der Benutzername und Einmalpasswort testweise an den SecOVID-Server geschickt werden können. Der SecOVID-Server und `tacping` müssen zur Verschlüsselung der abzusendenden Daten beide über den gleichen geheimen Schlüssel (secret key) verfügen. Zum Eintragen des Secret Keys im SecOVID-Server bearbeiten Sie bitte die Datei `tacacs`. Mit `tacping` stellen Sie dann folgendermaßen eine Passwortanfrage an den SecOVID-Server:

- `tacping -u username/password -S SecovidServerIP -K secretKey`
- z.B.`tacping -u smith/26011735 -S 192.168.1.1 -K secret`

Dabei ist `username` ein im SecOVID-Server registrierter Benutzer und `password` ein für diesen User gültiges Einmalpasswort.
In der Ausgabe von `tacping` bedeutet „Accept“, dass die Kombination Username, Passwort korrekt war, während „Rejected“ die Ablehnung des SecOVID-Servers anzeigt.

Ein RSA-Testclient: ovid_ping Das hier zu verwendende Testprogramm `ovid_ping` befindet sich nach der Installation des SecOVID-Servers im Verzeichnis `/etc/SecOvid`. Falls Sie das Testprogramm auf einem anderen Rechner in Ihrem Netzwerk laufen lassen möchten, so kopieren Sie bitte von Ihrer Installations-CD (`$UNIX/clients/ovid_ping` oder `\win32\clients\ovid_ping.exe`) auf die entsprechende Festplatte.
Bei `ovid_ping` handelt es sich um eine einfache RSA-Client-Software, mit der Benutzername und Einmalpasswort testweise an den SecOVID-Server geschickt werden können. Für eine korrekte Kommunikation zwischen `ovid_ping` und SecOVID-Server müssen auf dem Rechner, auf dem `ovid_ping` gestartet werden soll, an der vorgesehenen Stelle zwei Dateien mit geeignetem Inhalt vorhanden sein:

- Im Falle eines UNIX-Systems:
  `/etc/SecOvid/ovid_config` und `/etc/SecOvid/ovidcomm.pub`
- Im Falle von Windows 2000/XP/2003/Vista:
  `\Winnt\system32\drivers\etc\ovid_config` und
  `\Winnt\system32\drivers\etc\ovidcomm.pub`

Genaueres zu den beiden Dateien:

- Die Datei `ovid_config` muss dabei die IP-Adresse des SecOVID-Servers und die Portnummer, auf der dieser RSA-Anfragen entgegennimmt (vgl. den Eintrag hinter `secovid` in `/etc/services` auf dem Rechner des SecOVID-Servers), enthalten. IP-Adresse und Portnummer sind durch Doppelpunkt zu trennen. Auf Windows-Systemen muss die Datei mit einer leeren Zeile (newline) enden. Die Datei könnte folgenden Inhalt haben: `192.168.88.181:1647`
- In `ovidcomm.pub` muss der Public Key des SecOVID-Servers stehen.
  (Kopieren Sie ggf. die Datei `/etc/SecOvid/ovidcomm.pub` des SecOVID-Servers.)

Mit `ovid_ping` können Sie dann folgendermaßen Einmalpasswort-Anfragen an den SecOVID-Server schicken:
`/etc/SecOvid/ovid_ping -u username/password`
Dabei ist `username` ein im SecOVID-Server registrierter Benutzer und `password` ein für diesen User gültiges Einmalpasswort.
In der Ausgabe von `ovid_ping` bedeutet „Acked“ (acknowledged), dass die Kombination Username, Passwort korrekt war, während „Rejected“ die Ablehnung des SecOVID-Servers anzeigt. Die Ausgabe „Timed out“ zeigt Ihnen an, dass die gesicherte Kommunikation zwischen `ovid_ping` und dem SecOVID-Server nicht funktioniert hat, beispielsweise weil `ovid_ping` nicht - wie erforderlich - den Public Key des SecOVID-Servers verwendet hat oder weil der SecOVID-Server nicht erreichbar ist.

2.2 SecOVID-Backup- und Hochverfügbarkeitssystem*

Das SecOVID-Backupsystem und Hochverfügbarkeitssystem gewährleistet, dass auch im Störungsfall (Ausfall des SecOVID-Hauptservers) die aktuellen Benutzerdaten der SecOVID-Datenbank sowie ein SecOVID-Server ständig zur Verfügung stehen, so dass sich Benutzer jederzeit authentifizieren können.
Die Benutzerdaten eines als *SecOVID-Hauptserver* deklarierten Rechners können hierbei auf dem *SecOVID-Backupserver* redundant gehalten werden. Jede Modifikation der Benutzerdaten auf dem SecOVID-Hauptserver wird automatisch auf dem SecOVID-Backup-Server ausgeführt, so dass der SecOVID-Backupserver immer auf dem gleichen Datenbestand arbeitet wie der SecOVID-Hauptserver. Eine Modifikation der Benutzerdaten auf dem Hauptserver kann zwei Ursachen haben (siehe Abbildung 2.1):

- Erhält der SecOVID-Hauptserver eine Authentifikationsanfrage, so sendet der Hauptserver eine Kopie dieser Anfrage an seinen Backupserver weiter. Der Backupserver wird bei der Überprüfung der Anfrage zum gleichen Ergebnis gelangen wie der Hauptserver und entsprechend ggf. die gleiche Änderung am entsprechenden Datenbankeintrag vornehmen (evtl. unterscheidet sich allerdings der Zeitstempel für diesen Authentifizierungsversuch leicht von dem entsprechenden Zeitstempel auf dem Hauptserver, so dass die Dateien die die Datenbanken enthalten im allgemeinen nicht vollständig identisch sind). Der Backupserver sendet das Ergebnis seiner Überprüfung an den Hauptserver zurück.

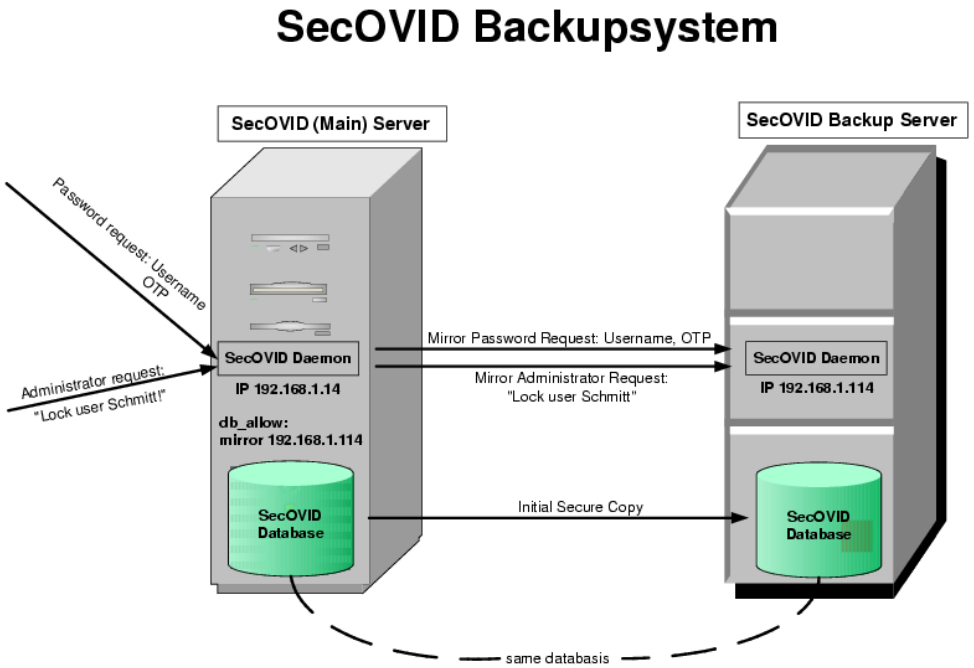


Abbildung 2.1: Datenredundanz (Backup) durch Spiegeln aller Anfragen

- Werden administrative Eingriffe auf dem SecOVID-Hauptserver vorgenommen (über eines der SecOVID-Administrationstools), so werden diese automatisch auch auf dem Backupserver durchgeführt.

Sie als verantwortlicher SecOVID-Administrator entscheiden, ob Sie es dabei belassen wollen, die Daten redundant zu halten, um beim Ausfall des Hauptservers den Backupserver manuell unter der IP-Adresse des Hauptservers zu starten. (Dies nennen wir *reines Backup-System* oder *Redundanz*, siehe Abbildung 2.1).

Oder Sie entscheiden sich dafür, *Hochverfügbarkeit* gewährleisten zu wollen, indem der Backupserver beim Ausfall des Hauptservers an dessen Stelle einspringt. Um Hochverfügbarkeit zu gewährleisten, haben Sie entweder auf dem Backupserver und auf dem Hauptserver einen weiteren Dienst zu installieren und zu starten (den *SecOVID-Backup-Dämon* `secbak`), oder Sie müssen allen Ihren Clients beide Server bekannt machen (sofern diese eine solche Konfigurationsmöglichkeit bieten).

Im ersten Fall übernimmt der Backupserver beim Ausfall des Hauptservers dessen IP-Adresse, so dass die anfragenden Rechner (z.B. Firewall) nur eine einzige IP-Adresse für den Authentifikationsserver kennen. Im zweiten Fall ist der Backupserver unter einer anderen IP-Adresse erreichbar und der anfragende Rechnern (z.B. Firewall) muss diese IP-Adresse kennen (z.B. durch Eintrag einer zweiten IP-Adresse und eines zweiten shared secret), um bei Nicht-Erreichbarkeit des Hauptservers den Backupserver befragen zu können.

Im folgenden erklären wir Ihnen zunächst detailliert die Funktionsweise des SecOVID-Backup- und Hochverfügbarkeitssystems. Dabei stellen wir Ihnen 3 verschiedene Strategien vor, unter denen Sie auswählen können. Anschließend erläutern wir, wie Sie für Datenredundanz sorgen (Installation des Backupsystems) oder Hochverfügbarkeit gewährleisten (Installation des Hochverfügbarkeitssystems und Konfiguration des Hochverfügbarkeitssystems). Schließlich beschreiben wir, wie Sie die vorher vorgestellten Strategien implementieren.

2.2.1 Funktionsweise

2.2.1.1 Welche Strategie ist für Sie die richtige?

Sie haben verschiedene Möglichkeiten, Redundanz (reines Backupsystem) und Hochverfügbarkeit zu realisieren. Im folgenden beschreiben wir die Hauptstrategien, die Sie verfolgen können. Auch andere als die nun vorgestellten Strategien lassen sich implementieren. (Konsultieren Sie bitte Abschnitt 2.2.3.)

Strategie 0: *Reines Backup - Clients kennen nur Hauptserver - kein Rollenwechsel, manueller IP-Wechsel im Problemfall*
Möchten Sie die SecOVID-Benutzerdaten redundant halten und im Falle eines Ausfalls des Hauptservers das Problem manuell beheben, z.B. indem Sie einem Backupserver manuell die IP-Adresse des Hauptservers zuweisen und den Backupserver erneut booten? (Allen SecOVID-Clients (z.B. Router, Firewalls) wäre lediglich ein SecOVID-Hauptserver bekannt, von der Existenz eines Backupservers wüssten sie nichts.)

Strategie 1: *Hochverfügbarkeit - Clients kennen Backupserver - Rollenwechsel, aber kein IP-Wechsel*
Möchten Sie die SecOVID-Benutzerdaten redundant halten und in allen SecOVID-Clients (z.B. Router, Firewalls) einen (oder mehrere) Backupserver für den Fall benennen, dass der Hauptserver keine Antwort liefert, so dass dann die Passwortanfragen an den Backupserver gesendet werden?

Strategie 2: *Hochverfügbarkeit - Clients kennen nur Hauptserver - Rollenwechsel, verbunden mit IP-Wechsel*
Möchten Sie die SecOVID-Benutzerdaten redundant halten und einen intelligenten Mechanismus installieren, der Ihnen Hochverfügbarkeit gewährleistet, ohne dass Sie manuell eingreifen müssen? Möchten Sie, dass dabei der Backupserver automatisch die Rolle des Hauptservers einnimmt und insbesondere (gelenkt durch ein KOBIL-Programm, welches im Hintergrund läuft) dessen IP-Adresse annimmt, sobald der SecOVID-Hauptserver nicht mehr verfügbar ist? Möchten Sie dabei, dass der ehemals als Hauptserver fungierende Rechner selbst bemerkt, dass dann bereits die Rolle des Hauptservers besetzt ist und daher seinerseits die Rolle des Backupservers einnimmt. (Allen SecOVID-Clients (z.B. Router, Firewalls) wäre lediglich ein SecOVID-Hauptserver bekannt, von der Existenz eines Backupservers wüssten sie nichts.)
Achtung: Diese Strategie wird nur unter Linux und Solaris unterstützt, nicht jedoch unter Windows.

2.2.1.2 Redundanz (reines Backupsystem)

Zur Verfolgung aller oben dargestellten Strategien haben Sie den Inhalt der SecOVID-Datenbank redundant zu halten. Hierzu veranlassen Sie den SecOVID-Hauptserver, künftig alle Passwort- und Administrationsanfragen auf den Backupserver zu spiegeln, indem Sie die IP-Adresse der SecOVID-Backupserver in der Datei `/etc/SecOvid/db_allow` auf dem designierten SecOVID-Hauptserver eintragen (mirror-Eintrag). Außerdem müssen sie sicherstellen, dass beide Rechner dieselbe Datei `/etc/SecOvid/ovidcomm.key` haben. Die genaue Vorgehensweise schildern wir in Abschnitt 2.2.2.1.
Beachten Sie, dass neben der datei `ovid_data`, die nun automatisch redundant gehalten wird auch eine ganze Reihe anderer Konfigurationsdateien vorhanden sind, etwa `users` oder `tacacs`. Diese sollten Sie nach evtl. Änderungen manuell auf das Backupsystem kopieren, um im Falle eines Ausfalls des Hauptservers alle Daten zur Verfügung zu haben, die Sie benötigen, um das Problem schnell zu beheben.

2.2.1.3 Hochverfügbarkeit (Strategie 1)

Wenn Sie zusätzlich zum reinen Backup auch noch eine automatische Nutzung des Backupservers im Falle des Ausfalls des Hauptservers ermöglichen wollen, können Sie einfach bei vielen SecOVID-Clients zwei Server konfigurieren (eben Haupt- und Backupserver). In diesem Fall wird der Client, wenn er vom ersten Server keine Antwort bekommt, die Anfrage automatisch an den zweiten Server richten. Um sich dagegen abzusichern, dass durch momentane Lastspitzen, die den Hauptserver zu Unrecht unerreichbar *scheinen* lassen, die Synchronität der beiden Server gefährdet wird, tragen sie in diesem Fall bitte zusätzlich in der Datei `/etc/SecOvid/db_allow` auf dem designierten Backup-Server einen mirror-Eintrag ein, der alle Anfragen auf den Hauptserver spiegelt, so dass nun beide Server alle Anfragen zum jeweils anderen spiegeln. Beachten Sie dabei, dass alle Konfigurationsdateien, z.B. `users`, `tacacs` und `clients` und insbesondere die Datenbank `ovid_data` auf beiden Servern

synchron sein müssen. Dies ist insbesondere beim initialen Starten des Systems und wenn einer der beiden Rechner nach einem Ausfall wieder hochgefahren wird, ein nicht ganz triviales Problem. Im laufenden Betrieb sorgt das automatische Spiegeln der Anfragen dafür, dass zumindest die Datenbank `ovid_data` auf beiden Rechnern synchron bleibt, für die sonstigen Konfigurationsdateien müssen Sie allerdings selbst daran denken, diese auf *beiden* Rechnern zu ändern und gegebenenfalls auf *beiden* Rechnern den SecOVID-Dämon anzuhalten und neu zu starten bzw. ihn durch Senden des `HUP`-Signals zu veranlassen, die Konfigurationsdateien neu einzulesen. Wenn einer der beiden Server neu gestartet wird, während der andere läuft, ist zu beachten, dass prinzipiell alle Dateien auf dem laufenden Server neuer sein könnten als auf dem, der gerade gestartet wird. Die Konfigurationsdateien lassen sich einfach kopieren, das Kopieren der Datei `ovid_data` ist allerdings ein potentielles Problem, da durch Passwort- oder Authentifikationsanfragen ja gerade zum Zeitpunkt des Kopierens eine Änderung der Datei veranlasst werden könnte. Für Windows-Rechner besteht die einzige Möglichkeit, sicherzustellen, dass ein konsistenter Stand der Datenbank übertragen wird, darin, den SecOVID-Dämon anzuhalten, die Datei `ovid_data` zu kopieren, und auf beiden Rechnern den SecOVID-Dämon möglichst gleichzeitig zu starten. Auf Linux- und Solaris-Systemen können Sie alternativ dem SecOVID-Dämon auf dem laufenden Rechner das Signal `USR1` schicken (dadurch werden Schreibzugriffe auf die Datei `ovid_data` blockiert), die Datei kopieren, und dann mit dem Signal `USR2` den Schreibzugriff wieder freigeben und den SecOVID-Dämon auf dem zweiten Rechner starten. Allerdings haben Sie dann das Problem, dass alle Änderungen der Datenbank, die zwischen dem Signal `USR1` und der Verfügbarkeit der Dienste des Dämons auf dem zweiten Rechner stattfinden, den zweiten Dämon nicht erreichen. Für Administrationsanfragen haben Sie aber selber in der Hand, es zu vermeiden, diese ausgerechnet in dieser kritischen Phase zu stellen und bei Passwortanfragen sollte es kein grosses Problem sein, wenn einzelne den zweiten Rechner nicht erreichen, da die Datenbank sich bei der nächsten Anmeldung eines solchen Benutzers ohnehin wieder synchronisieren wird, wenn nicht gerade zufällig die Suchtiefe überschritten wurde.

2.2.1.4 Hochverfügbarkeit (Strategie 2)

In diesem Unterabschnitt zeigen wir auf, dass das SecOVID-System über das reine Backup-System hinaus so konfiguriert werden kann, dass im Falle des Ausfalls des SecOVID-Hauptservers *automatisch* geeignete Aktionen veranlasst werden. Für diese Zwecke gibt es den SecOVID-Backup-Dämon (Serverprozess) `secbak`, der auf dem Hauptserver und auf dem Backupserver installiert wird. Mit dem SecOVID-Backup-Dämon lässt sich die oben vorgestellten Strategie 2 implementieren.
Hinweis: Der `secbak`-Dämon startet seinerseits den SecOVID-Dämon `secovid`. Stellen Sie also sicher, dass künftig nicht mehr der Dämon `secovid`, sondern statt dessen der Dämon `secbak` automatisch beim Booten des Rechners gestartet wird.
Das Verhalten des SecOVID-Backup-Dämons `secbak` hängt von den Einträgen der Konfigurationsdatei

```
/etc/SecOvid/secbak.conf
```

ab. Wir beschreiben im folgenden das Verhalten des SecOVID-Backup-Dämons auf dem Haupt- und dem Backupserver. Dabei gehen wir in drei Schritten vor:

- Zunächst beschreiben wir das Verhalten des SecOVID-Backup-Dämons beim Starten. Zunächst prüft der Rechner in einer Orientierungsphase, ob Hauptserver und/oder Backupserver bereits im Netz verfügbar sind. In Abhängigkeit von der Konfiguration und der Situation synchronisiert der SecOVID-Backup-Dämon zunächst die SecOVID-Datenbank, danach übernimmt der Rechner anschließend seine vorgesehene Rolle als Hauptserver oder Backupserver.

- Dann beschreiben wir die Rolle des Hauptservers und dessen Kontrolltätigkeit. (Der Hauptserver kontrolliert sich selbst und seinen Backupserver).

- Zuletzt beschreiben wir die Rolle des Backupservers und dessen Kontrolltätigkeit. (Der Backupserver kontrolliert sich selbst und seinen Hauptserver).

- **Der `secbak`-Dämon beim Start:** Beim Start sollte der Rechner die IP-Adresse `FirstIP` haben. Der `secbak`-Dämon betrachtet zunächst den Wert der Konfigurationsvariablen `BeServer`.

1. Steht diese auf 1, versucht der Dämon, ob er einen laufenden Hauptserver erreichen kann. Ist dies der Fall so hat der Backupserver zuvor nach einen Ausfall des Hauptservers die Rolle des Hauptservers übernommen, also wird der Rechner nun die Rolle des Backupservers übernehmen. Andernfalls übernimmt der Rechner jetzt die Rolle des Hauptservers. Hat die Konfigurationsvariable `ForceStart` dabei den Wert 1, wird sofort der SecOVID-Dämon gestartet, andernfalls wartet der Hauptserver zunächst auf den Backupserver und sobald dieser sich meldet, verschmelzen die beiden Rechner ihre lokalen Kopien der Datenbank `ovid_data` zu einer einheitlichen aktuellen Version, dann werden die SecOVID-Dämonen auf beiden Rechnern gestartet.
2. Steht `BeServer` auf 0, versucht der Dämon ebenfalls, einen schon laufenden Hauptserver zu kontaktieren. Schlägt dies fehl, beendet sich der SecOVID-Dämon. Ist der Hauptserver erreichbar, so wird die Konfigurationsvariable `Init` betrachtet, ist diese auf 1 gesetzt, so lädt der Backupserver die Konfigurationsdateien vom Hauptserver und leitet seine Konfiguration automatisch aus der des Hauptservers ab. Läuft auf dem Hauptserver bereits ein SecOVID-Dämon, so übernimmt der Backupserver nun einfach seine Datenbank und nimmt die Arbeit auf, andernfalls werden die Dateien `ovid_data` von Haupt- und Backupserver zu einer gemeinsamen Version verschmolzen, bevor beide Server die Arbeit aufnehmen.

- **Der `secbak`-Dämon auf dem Hauptserver:**

  1. Wenn der Hauptserver bemerkt, dass er keine Netzwerkverbindung in die Außenwelt hat, stoppt er seinen SecOVID-Dämon `secovid`. Zudem wechselt der Hauptserver – der nun nicht mehr als SecOVID-Hauptserver agiert – zur IP-Adresse `FirstIP`.
  2. Bei einer Unterbrechung der Netzwerkverbindung des Hauptservers zur Außenwelt wird der Backupserver die Aufgabe des Hauptservers übernehmen. Wird zu einem späteren Zeitpunkt die defekte Netzwerkverbindung wieder repariert, bleibt der ehemals als Hauptserver fungierende Rechner bei seiner Dummy-Adresse `FirstIP`, so dass er dem jetzt als Hauptserver agierenden Rechner nicht in die Quere kommt, aber dennoch für den Administrator für eine Reparatur via Remote Login auf die Adresse `FirstIP` zur Verfügung steht.
  3. Hat der Hauptserver Stromausfall, übernimmt der Backupserver die Rolle des Hauptservers, indem er dessen IP-Adresse annimmt. Wird der ehemals als Hauptserver fungierende Rechner zu einem späteren Zeitpunkt wieder hochgefahren, kommt es nicht zu Konflikten, wie wir jetzt sehen werden.

- **Der `secbak`-Dämon auf dem Hauptserver bei einem Reboot:**
  Wenn aus irgendwelchen Gründen der Hauptserver ausgefallen war, fährt der betreffende Rechner unter einer anderen IP-Adresse (Dummy-Adresse) hoch (unter der als `FirstIP` in der Datei `/etc/SecOvid/secbak.conf` definierten IP-Adresse). Der `secbak`-Dämon wird automatisch gestartet. Das Verhalten des `secbak`-Dämons geht aus der obigen Beschreibung hervor: Der `secbak`-Dämon kontrolliert, ob die Rolle des SecOVID-Hauptservers bereits besetzt ist, d.h. ob die IP-Adresse `OvidIP` bereits besetzt ist. (Details zu diesem Test: siehe Erklärungen zu `PingMode` und `CheckPort` in Abschnitt 2.2.3)
  Falls ja, prüft der Rechner, ob er die Rolle des Backupservers übernehmen darf, d.h. ob die IP-Adresse `BackupIP` noch nicht vergeben ist. Falls diese Adresse schon vergeben ist, liegt eine Fehlkonfiguration vor. Der Rechner bleibt in diesem Fall bei seiner Adresse `FirstIP` (er nimmt also weder die Rolle des Hauptservers noch die Rolle des Backupservers ein) und sendet eine Mail mit einer entsprechenden Warnung an den SecOVID-Administrator. Wenn die Rolle des Backupservers noch nicht vergeben ist, nimmt der Rechner nun dessen Rolle ein, wobei er zur IP-Adresse `BackupIP` wechselt.
  Die `FirstIP`-Adresse nimmt der Hauptserver also nach dem Booten ein, um in einer Orientierungsphase zu prüfen, ob bereits die Rolle des Hauptservers vergeben ist. Normalerweise ändert der Rechner nach dieser Orientierungsphase seine IP-Adresse, wie oben beschrieben:
  Falls die Rolle des SecOVID-Hauptservers bereits vergeben ist, weiß der `secbak`-Dämon, dass er einen Ausfall hatte und übernimmt die Rolle des Backupservers, wobei er zu dessen IP-Adresse wechselt. Falls die Rolle des SecOVID-Hauptservers noch nicht vergeben ist, übernimmt der Rechner nun die IP-Adresse des Hauptservers und wartet auf den Backupserver, um die Datenbanken zu verschmelzen oder startet den SecOVID-Dämon `secovid` (in Abhängigkeit von der Konfigurationsvariablen `|ForceStart|`.Zuvor

sorgt der `secbak`-Dämon für ein Kopieren der nun benötigten Dateien für SecOVID-Lizenz, `db_allow` und `clients`.

- **Der `secbak`-Dämon auf dem Backupserver:**
  Beim ersten Start des `secbak`-Dämons auf dem Backupserver werden alle notwendigen Dateien vom SecOVID-Hauptserver übernommen. Dann wird in einstellbaren Zeitintervallen kontrolliert, ob der SecOVID-Dämon auf dem Hauptserver läuft. Falls bei einer solchen Kontrolle festgestellt wird, dass der SecOVID-Dämon auf dem Hauptserver nicht läuft und der Rechner des Hauptservers netzwerktechnisch nicht erreichbar ist, nimmt der Backupserver fortan die Rolle des Hauptservers ein und wechselt dabei zur IP-Adresse des Hauptservers. Wenn der ehemals als Hauptserver fungierende Rechner anschließend wieder ans Netz kommt und bootet, merkt er, dass die Rolle des Hauptservers bereits vergeben ist und nimmt nun normalerweise seinerseits die Rolle des Backupservers ein (siehe oben).

- **Der `secbak`-Dämon auf dem Backupserver bei einem Reboot:**
  Beim Booten fährt der als Backupserver vorgesehene Rechner unter einer anderen IP-Adresse (Dummy-Adresse) hoch (unter der als `FirstIP` in der Datei `/etc/SecOvid/secbak.conf` definierten IP-Adresse), und der `secbak`-Dämon wird automatisch gestartet. Das Verhalten des `secbak`-Dämons geht aus obiger Beschreibung hervor: Der `secbak`-Dämon kontrolliert in einer Orientierungsphase, ob die Rolle des SecOVID-Hauptservers bereits besetzt ist, d.h. ob die IP-Adresse `OvidIP` bereits besetzt ist. (Details zu diesem Test: siehe Erklärungen zu `PingMode` und `CheckPort` in Abschnitt 2.2.3)
  Falls ja, prüft der Rechner, ob er die Rolle des Backupservers übernehmen darf, d.h. ob die IP-Adresse `BackupIP` noch nicht vergeben ist. Falls diese Adresse schon vergeben ist, liegt eine Fehlkonfiguration vor. Der Rechner bleibt in diesem Fall bei seiner Adresse `FirstIP` (er nimmt also weder die Rolle des Hauptservers noch die Rolle des Backupservers ein) und sendet eine Mail mit einer entsprechenden Warnung an den SecOVID-Administrator. Wenn die Rolle des Backupservers noch nicht vergeben ist (diese Situation wird jetzt normalerweise vorliegen), nimmt der Rechner nun dessen Rolle ein, wobei er zur IP-Adresse `BackupIP` wechselt. Zuvor besorgt sich der Rechner noch die aktuellen Benutzerdaten vom Hauptserver durch einen initialen Verschmelzungs- oder Kopiervorgang, je nachdem, ob der SecOVID-Dämon auf dem Hauptserver bereits läuft oder nicht. Falls der als Backupserver vorgesehene Rechner in seiner Orientierungsphase feststellt, dass die Rolle des Hauptservers noch nicht besetzt ist, benachrichtigt er den SecOVID-Administrator per Mail über den Ausfall des Hauptservers. Für eine initiale Besetzung der Rolle des Backupservers muss der Hauptserver verfügbar sein (für das initiale Kopieren der Benutzerdaten!). Daher behält der als Backupserver vorgesehene Rechner seine IP-Adresse `FirstIP`. Damit fungiert der Rechner nun weder als Hauptserver noch als Backupserver. Er stellt jede weitere Kontrolltätigkeit ein.

2.2.2 Installation

Hinweis: Die Installation des Backup-Systems ist nur auf Rechnern möglich, auf denen das SecOVID-Serversystem bereits mittels des Installationsskriptes `$UNIX/install.sh` von der SecOVID-CD installiert wurde.

2.2.2.1 Installation des Backupsystems

Zur Installation des SecOVID-Backupsystems (d.h. alle Benutzerdaten sollen redundant auf mindestens einem SecOVID-Backupserver gehalten werden) gehen Sie wie folgt vor:

- Loggen Sie sich als `root` auf dem Hauptserver und auf dem Backupserver ein.

- Zum Spiegeln aller an den Hauptserver gerichteten Authentifikations- und Administrationsanfragen, tragen Sie in der Datei `/etc/SecOvid/db_allow` auf dem SecOVID-Hauptserver die IP-Adresse des Backupservers ein:
  `mirror <IP-Adressen Backupserver>`
  Die Datei `/etc/SecOvid/db_allow` könnte beispielsweise so aussehen:

  ```
  allow 127.0.0.1
  ```

```
mirror 192.168.1.4
allow 192.168.1.113 192.168.1.114
192.168.1.115
```

In diesem Beispiel werden alle an den Hauptserver gerichteten Authentifikations- und Administrationsanfragen auf den Backupserver `192.168.1.4` gespiegelt. Administriert werden darf der Hauptserver nur vom lokalen Rechner aus (`127.0.0.1`) und von den Rechnern mit den IP-Adressen `192.168.1.113`, `192.168.1.114` und `192.168.1.115`.

- Erlauben Sie den Zugriff auf den Backupserver vom Hauptserver aus. Fügen Sie hierzu auf dem Backupserver in der Datei `/etc/SecOvid/db_allow` den Eintrag `allow <IP-Adresse SecOVID-Hauptserver>` hinzu (siehe oben).
- Stoppen Sie den SecOVID-Dämon `secovid` auf beiden Rechnern.
- In den Datenbanken von Haupt- und Backupserver müssen die gleichen Daten vorliegen. Existiert bereits eine SecOVID-Datenbank-Datei auf dem Hauptserver, muss diese auf den Backupserver kopiert werden. Dazu kopieren Sie auf Rechnern der gleichen Rechnerarchitektur einfach die Datei `ovid_data`.

  Verwenden Sie unterschiedliche Rechnerarchitekturen (etwa eine SUN SPARC und einen PC i86), so starten Sie den SecOVID-Server auf dem Hauptserver, markieren alle Benutzereinträge im Hauptserver mittels des SecOVID-Administrationstools (GUI) und exportieren diese. Ebenso verfahren Sie mit den Administratoren. Danach stoppen Sie den Server wieder, starten den SecOVID-Server auf dem Backupserver und importieren die entsprechende .db-Dateien auf dem Backupserver, um dann wiederum den SecOVID-Server zu stoppen.
- Machen Sie SecOVID-Hauptserver und SecOVID-Backupserver gegenseitig bekannt: Tragen Sie auf dem SecOVID-Hauptserver in der Datei `/etc/SecOvid/clients` IP-Adresse und shared secret des Backupservers ein (so, als würde der Backupserver RADIUS-Passwortanfragen an den Hauptserver stellen). Tragen Sie zudem auf dem SecOVID-Backupserver in der Datei `/etc/SecOvid/clients` die IP-Adresse und das shared secret des Hauptservers ein (so, als würde der Hauptserver RADIUS-Passwortanfragen an den Backupserver stellen). Kopieren Sie außerdem die Datei `/etc/SecOvid/ovidcomm.key` vom Hauptserver auf den Backupserver. Existiert diese Datei auf dem designierten Hauptserver noch nicht, so starten Sie den SecOVID-Dämon auf dem Hauptserver, warten darauf dass diese Datei erzeugt wird (dies kann mehrere Minuten dauern) und stoppen den SecOVID-Dämon wieder.
- Sorgen Sie darüber hinaus dafür, dass die nachfolgend aufgeführten Daten bzw. Dateien auf dem Hauptserver und den Backupservern exakt den gleichen Stand haben. Für gleichen Datenbestand können Sie z.B durch manuelles Kopieren der entsprechenden Dateien sorgen:
  - die RADIUS-Clients `/etc/SecOvid/clients`
  - die Profile der RADIUS-Benutzer `/etc/SecOvid/users`
  - die Profile der TACACS+-Benutzer `/etc/SecOvid/tacacs`
  - der Schlüssel für RSA-Clients `/etc/SecOvid/ovidcomm.pub`

Nach Abarbeitung der geschilderten Schritte und Neustart von Haupt- und Backupserver haben Sie das SecOVID-Backupsystem installiert und in Betrieb genommen. Das bedeutet, die SecOVID-Benutzerdaten des Hauptservers liegen ab jetzt automatisch redundant auf dem Backupserver vor. Von jetzt an leitet der SecOVID-Server nämlich alle Passwortanfragen und Administrationsanweisungen automatisch an den Backupserver weiter, welcher die Anfragen und Anweisungen genauso abhandelt wie der Hauptserver.
Achtung: Anfragen, die vom Backupserver ausgehend zum Hauptserver geschickt werden, werden *nicht* an den Backupserver zurückgespiegelt. Vermeiden Sie es daher am besten, das Administrationstool überhaupt auf dem Backupserver zu verwenden (evtl. mit Ausnahme des Einspielens/Aktualisierens der Datenbank wie oben beschrieben) oder einen Authentifikations-Client auf dem Backupserver zu installieren.

2.2.2.2 Installation des Hochverfügbarkeitssystems (Strategie 1)

Zur Installation der Hochverfügbarkeit (Ausfallsicherheit ohne manuellen Eingriff) des SecOVID-Systems müssen Sie entweder Haupt- und Backupserver und alle Clients geeignet konfigurieren (Strategie 1) oder den SecOVID-Backup-Dämon sowohl auf dem Rechner des Hauptservers als auch auf dem Rechner des Backupservers installieren (Strategie 2, siehe Abschnitt 2.2.2.3).
Für Strategie 1 gehen Sie vor, wie in Abschnitt 2.2.2.1 beschrieben. Zusätzlich tragen Sie auf dem Backupserver in der Datei `/etc/SecOvid/db_allow` auf dem IP-Adresse des Hauptservers ein:
`mirror <IP-Adressen Backupserver>`
Konfigurieren Sie zudem alle Clients so, dass Sie automatisch eine Authentifikationsanfrage an den Backupserver richten, falls der Hauptserver nicht antwortet. Im laufenden Betrieb beachten Sie die Hinweise aus Abschnitt 2.2.1.3.
Achtung: Anfragen, die von einem der beiden Server ausgehend zum anderen Server geschickt werden, werden *nicht* an den ursprünglichen Server zurückgespiegelt. Vermeiden Sie es daher, mit dem Administrationstools oder mit einem Authentifikations-Client von einem Rechner eine Anfrage an anderen Rechner zu erzeugen.

2.2.2.3 Installation des Hochverfügbarkeitssystems (Strategie 2)

Der SecOVID-Backup-Dämon ist zur Zeit unter folgenden Architekturen und Betriebssystemen verfügbar:

- PC i86/ SuSE Linux bzw. openSUSE 8.3 oder neuer
- SUN SPARC/ Solaris 9 oder neuer

Zur Installation des SecOVID-Backup-Dämons stellen Sie sicher, dass das SecOVID-Serversystem mittels des Installationsskriptes `$UNIX/install.sh` von der SecOVID-CD zuvor auf dem betreffenden Rechner installiert wurde.

Hinweis: Hauptserver und Backupserver müssen auf der gleichen Rechnerarchitektur laufen (beides PCs i86 oder beides SUN SPARC)!
Gehen Sie für die beiden Rechner, welche die Rolle des Hauptservers und des Backupservers übernehmen sollen, jeweils wie folgt vor:

- Loggen Sie sich als `root` ein.
- Kopieren Sie die SecOVID-Lizenzdateien für Hauptserver und Backupserver auf die Festplatte des designierten Hauptservers in das Verzeichnis `/etc/SecOvid/`.
- Falls dies nicht bereits automatisch geschehen ist, so mounten Sie Ihr CD-ROM-Laufwerk, so dass Programme auf der SecOVID-CD ausführbar sind: `mount -o exec /cdrom`
- Installieren Sie den SecOVID-Backup-Dämon `secbak` auf dem Hauptserver und dem Backupserver durch Aufrufen des Installationsskriptes `$UNIX/install_backupserver.sh` von der SecOVID-CD.
- Konfigurieren Sie mit Hilfe des mitinstallierten Programms `secbakconf` (X11 basiert) die beiden Rechner als Hauptserver bzw. als Backupserver.
- Automatisches Starten des SecOVID-Backup–Dämons `secbak`.
  Die Datei `/etc/SecOvid/secbak_start_script` ist das Startupskript für den `secbak`-Dämon. Kopieren Sie diese Datei in das Verzeichnis, in dem sich alle anderen Initskripte befinden (`/etc/rc.d/`) und setzen Sie einen Softlink auf `/etc/rc.d/rc*.d/S65secovid` (bzw. unter Solaris auf `/etc/rc*.d/S65secovid`). Löschen Sie den Link auf das Startskript oder die Datei zum Starten des `secovid`-Dämons (die Datei `S65secovid` in den Verzeichnissen `/etc/rc.d/rc*.d/` bzw. `/etc/rc*.d/`), denn der SecOVID-Dämon wird künftig vom `secbak`-Dämon gestartet.
- Booten Sie zunächst den Rechner, der die Rolle des Hauptservers einnehmen soll.
- Booten Sie dann den Rechner, der die Rolle des Backupservers einnehmen soll.

- Vergewissern Sie sich, dass der Hauptserver Passwort- und Administrationsanfragen korrekt abarbeitet und ggf. auf den Backupserver spiegelt. Hierzu legen Sie einen Benutzer mit Einmalpasswortgenerator mit dem SecOVID-Administrationstool (GUI) an (siehe Abschnitt 3.1.2) und senden Sie mit dem RADIUS-Testprogramm `radping` Passwortanfragen an den Hauptserver (siehe Abschnitt 2.1.7).
- Simulieren Sie Ausfälle des Hauptservers oder des Backupservers und überzeugen Sie sich, dass das System sich wie gewünscht verhält.

Achtung: Anfragen, die vom Backupserver ausgehend zum Hauptserver geschickt werden, werden *nicht* an den Backupserver zurückgespiegelt. Vermeiden Sie es daher am besten, das Administrationstool überhaupt auf dem Rechner zu verwenden, der gerade die Rolle des Backupservers innehat oder einen Authentifikations-Client auf diesem Rechner zu benutzen.

2.2.3 Konfiguration des Hochverfügbarkeitssystems

Nach Starten des Installationsskriptes für das SecOVID-Hochverfügbarkeitssystem
`$UNIX/install_backupserver.sh` von der SecOVID-CD enthält das Verzeichnis `/etc/SecOvid` insbesondere die Dateien `secbak`, `secbak.conf` und `secbak_start_script`. Details zu diesen drei Dateien erläutern wir im Folgenden.

2.2.3.1 Das Executable secbak

Die Datei `secbak` ist der SecOVID-Backup-Dämon, der normalerweise auf den beiden Rechnern installiert wird, die als SecOVID-Hauptserver und als SecOVID-Backupserver dienen sollen. Der `secbak`-Dämon lädt nach dem Starten die Konfigurationsdatei `/etc/SecOVID/secbak.conf`.

2.2.3.2 Das Startskript

Die Datei `secbak_start_script` ist das Startupskript für den `secbak`-Dämon. Kopieren Sie diese Datei in das Verzeichnis, in dem sich alle anderen Initskripte befinden (`/etc/rc.d/`) und setzen Sie einen Softlink auf `/etc/rc.d/rc*.d/S65secovid` (bzw. unter Solaris auf `/etc/rc*.d/S65secovid`). Löschen Sie den Link auf das Startskript oder die Datei zum Starten des `secovid`-Dämons (die Datei `S65secovid` in den Verzeichnissen `/etc/rc.d/rc*.d/` bzw. `/etc/rc*.d/`), denn der SecOVID-Dämon wird künftig vom `secbak`-Dämon gestartet.

2.2.3.3 Die Konfigurationsdatei secbak.conf

Der SecOVID-Backup-Dämon lädt nach dem Starten seine Konfigurationsdatei `/etc/SecOvid/secbak.conf`. Über diese Datei teilen Sie dem Rechner beispielsweise mit, ob er normalerweise als Hauptserver oder Backupserver fungieren soll. Beachten Sie, dass Hauptserver und Backupserver unterschiedlich konfiguriert werden müssen, also unterschiedliche Dateien `secbak.conf` besitzen müssen.
Die Bearbeitung der Dateien `secbak.conf` können Sie im wesentlichen mittels des GUI-Programms `secbakconf` vornehmen, im folgenden wollen wir am Beispiel einer Konfigurationdatei `secbak.conf`, wie sie auf einem Rechner vorliegen könnte, der normalerweise die Rolle des Hauptservers einnehmen soll, einige Hinweise zur manuellen Konfiguration für Notfälle geben und einige Details erläutern, die über die Möglichkeiten des Konfigurationsprogramms hinausgehen. Die entsprechende Datei finden Sie nach Installation des SecOVID-Backup-Dämons unter `/etc/SecOvid/secbak.conf_mainserver_strategy3`. Das Beispiel einer passenden Konfigurationsdatei für einen Backupserver finden Sie unter
`/etc/SecOvid/examples_secbak/secbak.conf_backupserver_strategy3`. (Schauen Sie sich auch die zu den anderen Hauptstrategien gehörenden Konfigurationsdateien an:
`/etc/SecOvid/examples_secbak/secbak.conf_mainserver_strategy*` für die als Hauptserver vorgesehenen Rechner und
`/etc/SecOvid/examples_secbak/secbak.conf_backupserver_strategy*` für die als Backupserver vorgesehenen Rechner.)

```
BeServer=1   # You are configured to be the main SecOVID server
Strategy=2   # Codes the strategy to be used.
             # Strategy=2 means on backup server: Change IP if there are problems
SSLCopy=0    # Make a SSL Backup
FirstIP=192.168.1.252 # The IP we use when booting (dummy address)
SecondIP=192.168.1.14 # The IP we change to after (working address)
OvidIP=192.168.1.14   # The IP of the SecOVID main server
BackupIP=192.168.1.114  # The IP of the SecOVID backup server
ServerLicence=/etc/SecOvid/ovid_licence_server
BackupLicence=/etc/SecOvid/ovid_licence_backup
CheckIP=192.168.1.2  # The IP used for the online test
PingMode=0           # The method used for availabilty tests
CheckPort=22         # The port of CheckIP used for availability tests (if PingMode=1)
CheckTime=3          # Some type of sleep time
WaitTime=3           # Number of tries made for checking availability of some computer
StartOvid=bash /etc/rc.d/secovid
ChangeServerIP=ifconfig eth0
RestartRouting=route add default gw 192.168.1.1
MailSend=bash /etc/SecOvid/Alert.sh
CopyUserProfiles=1      # Copy the RADIUS file 'users' and the TACACS+ file 'tacacs'
BackupHostname=hostname bluenote  # Host name of backup server
OvidHostname=hostname blue    # Host name of main server
BackupPort=4715      # Portnumber used for communication between secbak daemons
ClientsServer=/etc/SecOvid/clients_server  # The clients file to be used for main server
ClientsBackup=/etc/SecOvid/clients_backup  # The clients file to be used for backup server
DballowServer=/etc/SecOvid/db_allow_server # The db_allow file to be used for main server
DballowBackup=/etc/SecOvid/db_allow_backup # The db_allow file to be used for backup server
```

Allgemein können Kommentare mit `#` deklariert werden. Die im Beispiel aufgeführten Einträge haben folgende Bedeutung:

- `BeServer`: Diese Variable darf die Werte 1 und 0 haben und legt fest, ob der vorliegende Rechner normalerweise als Hauptserver (dann setzen Sie den Wert 1) oder als Backupserver (dann setzen Sie den Wert 0) dienen soll.
- `Strategy`: Auf dem als Hauptserver vorgesehenen Rechner (`BeServer=1`) wird diese Variable ignoriert. Auf dem als Backupserver vorgesehenen Rechner gibt die Variable aus, welche der oben vorgestellten Strategien (1 oder 2) angewendet werden soll. Derzeit sollte hier immer der Wert 2 gewählt werden.
- `SSLCopy`: Diese Variable muss immer den Wert 0 haben.
- `FirstIP`: Das ist die IP-Adresse, unter der der Rechner hochfährt, um in seiner Orientierungsphase zu prüfen, ob die Rolle des Hauptservers bereits besetzt ist, d.h. ob der SecOVID-Dämon `secovid` unter der IP-Adresse `OvidIP` läuft. Die `FirstIP`-Adresse des als Hauptservers und des als Backupservers vorgesehenen Rechners müssen in die durch `ClientsServer` und `ClientsBackup` spezifizierten Dateien auf Hauptserver und Backupserver eingetragen werden (d.h. Eintragen von jeweils zwei IP-Adressen in vier verschiedene Dateien).
- `SecondIP`: Das ist die IP-Adresse, die der Rechner normalerweise nach der Orientierungsphase annimmt. Für den als Hauptserver vorgesehenen Rechner (`BeServer=1`) tragen Sie hier die unter `OvidIP` definierte Adresse ein, für den als Backupserver vorgesehenen Rechner (`BeServer=0`) tragen Sie bitte die unter `BackupIP` definierte Adresse ein. Die `SecondIP`-Adresse des als Hauptservers und des als Backupservers vorgesehenen Rechners müssen in die durch `ClientsServer` und `ClientsBackup` spezifizierten Dateien auf Hauptserver und Backupserver eingetragen werden (d.h. Eintragen von jeweils zwei IP-Adressen in vier verschiedene Dateien).

- `OvidIP`: Die IP-Adresse, die der Rechner besitzt, der die Rolle des Hauptservers übernommen hat.
- `BackupIP`: Die IP-Adresse, die der Rechner besitzt, der die Rolle des Backupservers übernommen hat.
- `ServerLicence`: Die SecOVID-Lizenzdatei des Hauptservers, welche die IP-Adresse `OvidIP` lizensiert.
- `BackupLicence`: Die SecOVID-Lizenzdatei des Backupservers, welche die IP-Adresse `BackupIP` lizensiert. Die Lizenzdateien von Hauptserver und Backupserver müssen verschieden sein.
- `CheckIP`: Die IP-Adresse des Rechners, mit dem der vorliegende Rechner zu kommunizieren versucht, um zu prüfen, ob er selbst (als Hauptserver oder als Backupserver) online (d.h. im Netz sichtbar) ist. Im Falle, dass der vorliegende Rechner nicht online ist, kann er die Adresse `CheckIP` nicht erreichen. Als `CheckIP` sollten Sie die IP-Adresse eines Rechners eintragen, der bei einer defekten Netzwerkverbindung zum vorliegenden Rechner (Hauptserver oder Backupserver) nicht erreichbar ist. Sie könnten als `CheckIP` beispielsweise die IP-Adresse des Gateways eintragen.
- `PingMode`: Hier sollte der Wert 0 stehen. Dies bedeutet, dass für den oben geschilderten Online-Test (sowie für den Test, ob andere Rechner netzwerktechnisch verfügbar sind) versucht wird, mittels `ping` mit dem Rechner `CheckIP` zu kommunizieren. In manchen Netzwerkumgebungen scheitert dieser Test, da ein ping durch eine Firewall unterbunden wird, obwohl eine Netzwerkverbindung vorhanden ist. In diesem Fall setzen Sie `PingMode=1`. Dann versucht der vorliegende Rechner beim Online-Test, eine Verbindung zum Port `CheckPort` des Rechners `CheckIP` herzustellen.
- `CheckPort`: Im Falle `PingMode=0` wird dieser Wert ignoriert. Im Falle `PingMode=1` versucht der Rechner, den Computer `CheckIP` (bzw. einen sonstigen Rechner) auf Port `CheckPort` anzusprechen, um zu prüfen, ob er selbst online ist.
- `CheckTime`: Das Zeitintervall in Sekunden, in welchem der Backupserver kontrolliert, ob der Hauptserver läuft und netzwerktechnisch vom Backupserver erreichbar ist. Gleichzeitig das Zeitintervall in Sekunden, in dem der Hauptserver kontrolliert, ob er selbst online (d.h. netzwerktechnisch erreichbar) ist.
- `WaitTime`: Die Anzahl der Versuche, die der vorliegende Rechner bei einem Erreichbarkeitstest unternimmt, ehe der vorliegende Rechner der Meinung ist, der SecOVID-Hauptserver sei ausgefallen. Das heißt beispielsweise, dass der Backupserver nach `CheckTime`*`WaitTime` Sekunden die Rolle des Hauptservers einnimmt.
- `StartOvid`: Der Befehl zum Starten des SecOVID-Dämons `secovid`. Diesen Eintrag müssen Sie normalerweise nicht verändern.
- `ChangeServerIP`: Der Befehl zum Wechsel der IP-Adresse. Wenn der Backupserver die IP-Adresse des Hauptservers im Falle des Ausfalls des Hauptservers übernehmen soll, wird dieses Kommando für den Wechsel der IP-Adresse verwendet.
  Achtung: Beachten Sie, dass Sie auch bei Verwendung des Konfigurationsprogramms `secbakconf` hier nach dem ersten Start des Backupservers, nachdem der Backupserver seine Konfiguration aktualisiert hat, eine manuelle Anpassung des Namens der Netzwerkschnittstelle vornehmen müssen, falls dieser Name nicht für Haupt- und Backupserver derselbe ist!
- `RestartRouting`: Der Befehl zum Erstellen des Routing. Generell geht das Routing verloren, wenn man die IP-Adresse des Rechners umsetzt. Mittels dieses Befehls wird das Routing nach jedem Wechsel der IP-Adresse des vorliegenden Rechners repariert.
- `MailSend`: Das Kommando, mit dem automatisch Mails an den SecOVID-Administrator gesendet werden, um ihn über Unregelmäßigkeiten bzgl. der Verfügbarkeit von Haupt- oder Backupserver zu unterrichten. (Dies ist natürlich nur möglich, falls prinzipiell Mails vom vorliegenden Rechner aus geschickt werden können.) Der Defaulteintrag `bash /etc/SecOvid/Alert.sh` ist ein Beispiel. Hier wird für die Benachrichtigung des Administrators das Skript `/etc/SecOvid/Alert.sh` ausgeführt. Sie können hier noch andere Befehle in das Skript einfügen.

- `CopyUserProfiles`: Diese Variable darf die Werte 0 oder 1 haben. Im Fall `CopyUserProfiles=1` werden die Dateien `/etc/SecOvid/users` und `/etc/SecOvid/tacacs` vom vorliegenden Rechner alle `CheckTime` Sekunden auf den Backupserver kopiert, sofern der vorliegende Rechner die Rolle des Hauptservers besetzt.
- `BackupHostname`: Mit diesem Kommando wird der Hostname des Backupservers (Rechner mit der IP-Adresse `BackupIP`) gesetzt. Dieses Kommando wird durch den Backup-Dämon `secbak` aufgerufen, wenn der vorliegende Rechner die Rolle des Backupservers einnimmt.
- `OvidHostname`: Mit diesem Kommando wir der Hostname des Hauptservers (Rechner mit der IP-Adresse `OvidIP`) gesetzt. Dieses Kommando wird durch den Backup-Dämon `secbak` aufgerufen, wenn der vorliegende Rechner die Rolle des Hauptservers einnimmt.
- `BackupPort`: Die Portnummer, über welche die Backup-Dämons `secbak` miteinander kommunizieren. Setzen Sie auf den Rechnern, die als Hauptserver und Backupserver dienen sollen, die gleiche Portnummer!
- `ClientsServer`: Die `clients`-Datei des Hauptservers. Nimmt der vorliegende Rechner die Rolle des Hauptservers an, wird auf dem Rechner ein Softlink von `ClientsServer` nach `/etc/SecOvid/clients` gesetzt.
- `ClientsBackup`: Die `clients`-Datei des Backupservers. Nimmt der vorliegende Rechner die Rolle des Backupservers an, wird auf dem Rechner ein Softlink von `ClientsBackup` nach `/etc/SecOvid/clients` gesetzt.
- `DballowServer`: Die `db_allow`-Datei des Hauptservers. Nimmt der vorliegende Rechner die Rolle des Hauptservers an, wird auf dem Rechner ein Softlink von `DballowServer` nach `/etc/SecOvid/db_allow` gesetzt.
- `DballowBackup`: Die `db_allow`-Datei des Backupservers. Nimmt der vorliegende Rechner die Rolle des Backupservers an, wird auf dem Rechner ein Softlink von `DballowBackup` nach `/etc/SecOvid/db_allow` gesetzt.
- `ForceStart`: Dieser zusätzliche Parameter kann auf den Wert 1 gesetzt werden, um den Start des Hauptserver zu erzwingen, auch wenn der Backupserver nicht erreichbar ist.
  Achtung: Ohne diesen Parameter bzw. wenn er auf den Defaultwert 0 eingestellt ist, wartet der Hauptserver vor dem Start stets auf den Backupserver um die Datenbank zu synchronisieren. Im Falle eines Hardwaredefekts müssen Sie also diesen Parameter von Hand umsetzen und ihn, wenn wieder beide Rechner verfügbar sind, auch wieder von Hand zurücksetzen.

2.2.3.4 Implementierung der verschiedenen Strategien

Strategie 0: *Reines Backup - Clients kennen nur Hauptserver - kein Rollenwechsel, manueller IP-Wechsel im Problemfall*
Wir nehmen an, Sie wollen Ihr SecOVID-System wie folgt konfigurieren:
Sie möchten die SecOVID-Benutzerdaten redundant auf einem Backupserver halten. Im Falle eines Ausfalls des Hauptservers möchten Sie das Problem manuell beheben, z.B. indem Sie einem Backupserver manuell die IP-Adresse des Hauptservers zuweisen und den Backupserver erneut booten. Sie sorgen selbst dafür, dass der ehemals als Hauptserver dienende Rechner sich anschließend nicht mehr unter der gleichen IP-Adresse meldet, die ja nun der ursprünglich als Backupserver dienende Rechner besetzt. Allen SecOVID-Clients (z.B. Router, Firewalls) ist lediglich ein SecOVID-Hauptserver bekannt. Von der Existenz eines Backupservers wissen die SecOVID-Clients nichts. Um die geschilderte Situation zu realisieren, gehen Sie wie folgt vor:

1. Installieren Sie (falls noch nicht geschehen) den SecOVID-Dämon `secovid` auf den beiden Rechnern, die Hauptserver bzw. Backupserver werden sollen. (Hierzu verwenden Sie das Skript `$UNIX/install.sh` von Ihrer SecOVID-CD.)

2. Sorgen Sie für Redundanz der SecOVID-Benutzerdaten, indem Sie auf dem Hauptserver entsprechende mirror-Einträge setzen und die Datei `ovidcomm.key` kopieren, so dass künftig alle an den Hauptserver gerichteten Passwort- und Administrationsanfragen auf den Backupserver gespiegelt werden (siehe Abschnitt 2.2.2.1).
3. Beenden Sie die SecOVID-Dämons `secovid` auf dem designierten Haupt- und dem designierten Backupserver.
4. Starten Sie dann die SecOVID-Dämons erneut auf beiden Rechnern.

Strategie 1: *Hochverfügbarkeit - Clients kennen Backupserver - Rollenwechsel, aber kein IP-Wechsel*
Wir nehmen an, Sie wollen Ihr SecOVID-System wie folgt konfigurieren:
Sie möchten die SecOVID-Benutzerdaten redundant auf einem Backupserver halten. Zudem benennen Sie in allen SecOVID-Clients (z.B. Router, Firewalls) den Backupserver, an den Passwortanfragen gesendet werden, für den Fall, dass der Hauptserver keine Antwort liefert. Der Backupserver soll ebenso wie der Hauptservers alle Anfragen beantworten können, ohne dazu seine IP-Adresse wechseln zu müssen.
Um die geschilderte Situation zu realisieren, gehen Sie wie folgt vor:

1. Installieren Sie (falls noch nicht geschehen) den SecOVID-Dämon `secovid` auf den beiden Rechnern, die Hauptserver bzw. Backupserver werden sollen. (Hierzu verwenden Sie das Skript `$UNIX/install.sh` von Ihrer SecOVID-CD.)
2. Sorgen Sie für Redundanz der SecOVID-Benutzerdaten, indem Sie auf Haupt- und Backupserver entsprechende mirror-Einträge setzen, so dass künftig alle Passwort- und Administrationsanfragen, die einen der beiden Rechner erreichen auf den anderen Server gespiegelt werden (siehe Abschnitt 2.2.2.1).
3. Machen Sie bei allen SecOVID-Clients den Backupserver als zweiten Authentifikationsserver bekannt (Eintragen der IP-Adresse und eines shared secret, siehe Abschnitt 3.2.2).
4. Registrieren Sie bei beiden Servern alle SecOVID-Clients (Eintragen der IP-Adresse und des shared secret, siehe Abschnitt 3.1.1).
5. Beenden Sie die SecOVID-Dämons `secovid` auf dem designierten Haupt- und dem designierten Backupserver.
6. Starten Sie dann den SecOVID-Backup-Dämon `secbak` zunächst auf dem Hauptserver, anschließend auf dem Backupserver oder booten Sie die beiden Rechner, zunächst den Hauptserver, dann den Backupserver.

Strategie 2: *Hochverfügbarkeit - Clients kennen nur Hauptserver - Rollenwechsel, verbunden mit IP-Wechsel*
Wir nehmen an, Sie wollen Ihr SecOVID-System wie folgt konfigurieren:
Sie möchten die SecOVID-Benutzerdaten redundant auf einem Backupserver halten. Zudem wollen Sie einen intelligenten Mechanismus installieren, der Ihnen Hochverfügbarkeit gewährleistet, ohne dass Sie manuell eingreifen müssen. Sobald der SecOVID-Hauptserver nicht mehr verfügbar ist, soll dabei der Backupserver automatisch die Rolle des Hauptservers einnehmen und insbesondere (gelenkt durch ein KOBIL-Programm, welches im Hintergrund läuft) dessen IP-Adresse annehmen. Der ehemals als Hauptserver fungierende Rechner soll (sobald er wieder netzwerktechnisch sichtbar ist) selbst bemerken, dass nun bereits die Rolle des Hauptservers besetzt ist. Daher soll er jetzt seinerseits die Rolle des Backupservers einnehmen. Allen SecOVID-Clients (z.B. Router, Firewalls) ist lediglich ein SecOVID-Hauptserver bekannt. Von der Existenz eines Backupservers wissen die SecOVID-Clients nichts.
Um die geschilderte Situation zu realisieren, gehen Sie wie folgt vor:

1. Installieren Sie (falls noch nicht geschehen) den SecOVID-Dämon `secovid` auf den beiden Rechnern, die Hauptserver bzw. Backupserver werden sollen. (Hierzu verwenden Sie das Skript `$UNIX/install.sh` von Ihrer SecOVID-CD.)
2. Sorgen Sie für Redundanz der SecOVID-Benutzerdaten, indem Sie auf dem Hauptserver entsprechende mirror-Einträge setzen, so dass künftig alle an den Hauptserver gerichteten Passwort- und Administrationsanfragen auf den Backupserver gespiegelt werden (siehe Abschnitt 2.2.2.1).

3. Installieren Sie das SecOVID-Hochverfügbarkeitssystem (Starten von `$UNIX/install_backupserver.sh` auf designiertem Hauptserver und designiertem Backupserver, siehe Abschnitt 2.2.2.3).
4. Nehmen Sie die miteels des Konfigurationstools `/etc/SecOvid/secbakconf` auf beiden Rechner die notwendigen Konfigurationseinträge in den Dateien `/etc/SecOvid/secbak.conf` vor.
5. Beenden Sie die SecOVID-Dämons `secovid` auf dem designierten Haupt- und dem designierten Backupserver.
6. Starten Sie dann den SecOVID-Backup-Dämon `secbak` zunächst auf dem Hauptserver, anschließend auf dem Backupserver oder booten Sie die beiden Rechner, zunächst den Hauptserver, dann den Backupserver.

2.3 Clientseite

Im folgenden beschreiben wir, wie man verschiedene Clients für SecOVID installiert. Unter SecOVID-Clients verstehen wir Software- oder Hardwaresysteme, die Passwortanfragen über eine der drei dem SecOVID-Server bekannten Schnittstellen (RADIUS, TACACS+ und RSA) weiterleiten können. SecOVID-Clients sind beispielsweise

- RADIUS-Server
- Firewalls
- Router
- VPN-Gateways (Virtual Private Networks) mit RADIUS- oder TACACS+-Interface
- Apache-Webserver mit dem KOBIL-eigenen RADIUS-Authentifikationsmodul,
- die KOBIL-eigenen PAM-Module mit RSA-Schnittstelle zum Schutz von UNIX-Netzen

Die mit „*“ markierten Komponenten (SecOVID-Clients) sind zum Minimalbetrieb des SecOVID-Systems nicht notwendig. Installieren Sie daher nur die SecOVID- Clients für die Zielsysteme, deren Benutzeridentifikation Sie verbessern möchten.

2.3.1 Installation eines RADIUS-Servers*

Falls Sie einen RADIUS-Server eines anderen Herstellers (RADIUS in Version 2) installieren möchten, so folgen Sie bitte den Instruktionen im Handbuch bzw. der Dokumentation Ihres RADIUS-Servers. Beachten Sie dabei bitte, dass Sie den RADIUS-Server wegen technischer Einschränkungen des RADIUS-Systems möglicherweise nicht auf dem gleichen Rechner installieren können wie den SecOVID-Server. An dieser Stelle sei erneut darauf hingewiesen, dass der SecOVID-Server die Funktionalität eines RADIUS-Servers bereits beherrscht. Daher müssen Sie nicht notwendigerweise einen separaten RADIUS-Server installieren.
Allerdings besitzt der SecOVID-Server sein eigenes proprietäres Accounting. Für ein RADIUS-konformes Accounting müssen Sie also doch einen separaten RADIUS-Server installieren, der Accounting unterstützt, z.b. mit **FreeRADIUS** (vgl. auch Abschnitt 2.1.4. In diesem Fall konfigurieren Sie den anderen Server so, dass er einerseits Accounting-Daten sammelt und andererseits die eigentlichen RADIUS-Requests über die Proxy-Schnittstelle an den SecOVID-Server weiterleitet. Im Falle von z.B. **FreeRADIUS** ist dies sogar möglich, ohne das ein zusätzlicher Rechner benötigt wird.
Nach der Installation des RADIUS-Servers müssen Sie RADIUS-Server und SecOVID-Server gegenseitig bekannt machen. Die notwendigen Konfigurationsschritte sind in den Abschnitten 3.1.1 und 3.2.1 beschrieben.

2.3.2 Installation eines RADIUS-Clients*

Falls Sie den SecOVID-Einmalpasswortmechanismus in irgendeinem Produkt anwenden wollen, welches über eine RADIUS-Schnittstelle (RADIUS Version 2) verfügt, so installieren Sie das betreffende Produkt und tragen Sie an der vorgesehenen Stelle die IP-Adresse des SecOVID-Servers (ggf. als RADIUS-Server) und ein shared secret ein (siehe Abschnitte 3.2.2 und 3.1.1). Analog müssen Sie den RADIUS-Client beim SecOVID-Server eintragen. Die notwendigen Konfigurationsschritte sind in den Abschnitten 3.1.1 und 3.2.1 beschrieben.

2.3.3 Installation eines TACACS+-Clients*

Falls Sie den SecOVID-Einmalpasswortmechanismus in irgendeinem Produkt anwenden wollen, welches über eine TACACS+-Schnittstelle verfügt, so installieren Sie das betreffende Produkt und tragen Sie an der vorgesehenen Stelle die IP-Adresse des SecOVID-Servers (ggf. als TACACS+-Server) und ein shared secret ein (siehe Abschnitte 3.2.3 und 3.1.1). Analog müssen Sie den TACACS+-Client beim SecOVID-Server eintragen. Die notwendigen Konfigurationsschritte sind in den Abschnitten 3.1.1 und 3.2.3 beschrieben.

2.3.4 Installation einer Firewall*

Falls Sie den SecOVID-Einmalpasswortmechanismus verwenden wollen, um eine Firewall sicherer zu machen, so muss die Firewall über eine RADIUS-Schnittstelle (RADIUS Version 2) oder eine TACACS+-Schnittstelle verfügen. Zur Installation der Firewall folgen Sie bitte den Instruktionen im Handbuch bzw. in der Dokumentation Ihrer Firewall. Informationen zur geeigneten Konfiguration der Firewall finden sie in Abschnitt 3.2.4. Für eine korrekte Kommunikation zwischen Firewall und SecOVID-Server müssen Sie zudem die Firewall beim SecOVID-Server als Client anmelden (siehe Abschnitt 3.1.1).

2.3.5 Installation: Schutz der Einwahl über Windows 2000/XP/2003 RAS*

Falls Sie den SecOVID-Einmalpasswortmechanismus verwenden wollen, um die Einwahl eines entfernten Clients in ein Windows 2000/XP/2003-Netz sicherer zu machen, so können Sie dies leicht mit dem Routing and Remote Access Service (RRAS) erreichen.
Zur Nutzung der in Windows 2000/XP RAS vorhandenen RADIUS-Schnittstelle konsultieren Sie bitte die entsprechende Dokumentation des Microsoft RAS-Servers.

2.3.6 Installation des Apache-Webservers mit SecOVID-Authentifikationsmodul*

Zur Installation des Apache-Webservers mit dem notwendigen SecOVID-Authentifikationsmodul starten Sie das Installationsskript `$UNIX/apache_install.sh` von der SecOVID-CD. (Sie können auch das SecOVID-Authentifikationsmodul `mod_auth_secovid.so` aus dem Archiv eigenständig in Ihren Apache-Server einbinden, siehe Abschnitt 3.2.6.)

Wenn Sie unser Authentifikationsmodul in einen vorhandenen Webserver integrieren wollen, finden Sie das Modul unter `$UNIX/apache/module_only`. Bitte beachten Sie auch das README in diesem Verzeichnis.
Im Abschnitt 3.2.6 finden Sie die notwendigen Informationen, um den Zugriff auf einzelne Verzeichnisse des Apache-Webservers dem SecOVID-Schutz zu unterstellen und somit den Zugriff auf einzelne Webseiten zu schützen.

2.3.7 Installation des Internet Information Server (IIS) mit SecOVID-Authentifikationsmodul*

Der Microsoft Internet Information Server (IIS) sollte immer zusammen mit dem Microsoft ISA Server oder einer vergleichbaren Firewall als Schnittstelle zum Internet betrieben werden.
In Abschnitt 2.3.8 ist im Detail beschrieben, wie der ISA Server mit SecOVID abgesichert werden kann. Falls Sie den ISA Server 2006 (oder neuer) verwenden, können Sie die sogenannte Kerberos Constraint Delegation verwenden, um vom Benutzer lediglich den Usernamen und das OTP abzufragen, und dennoch IIS-seitig eine

komplette Windows Authentisierung vorliegen zu haben. Fragen Sie nach dem KOBIL Integration Guide für den Microsoft ISA Server.

2.3.8 Installation des Internet Security and Acceleration Server mit SecOVID-Authentifikationsmodul*

Der Microsoft ISA Server 2006 unterstützt RADIUS bereits von Hause aus, bitte fordern Sie den ISA Server Integration Guide bei KOBIL an. Mit dieser Version des ISA Servers ist auch die Kerberos Constraint Delegation verfügbar, mit der eine externe Authentisierung nur mit Username/OTP möglich ist, also ohne Eingabe des Windows Passworts.

2.3.9 Installation und Konfiguration: Schutz eines UNIX-Netzes*

Sie können an allen Stellen, an denen bisher UNIX-Systeme Passwortabfragen zur Benutzerauthentifikation gemacht haben, statische UNIX-Passwörter durch Einmalpasswörter ersetzen oder ergänzen. Auf diese Weise kann die Passwortabfrage insbesondere bei den folgenden Anwendungen modifiziert werden:
lokales login an der Konsole und damit auch telnet, rsh und (teilweise) rlogin, ferner rlogin, xdm und xscreensaver oder xlock. Beachten sie, das nicht alle Programme in allen UNIX-Distributionen PAM unterstützen, Sie müssen also im Einzelfall (inbesondere bei X-basierten Programmen) ausprobieren, ob das gewünschte Programm funktioniert und gegebenenfalls nach einem alternativen Programm mit vergleichbarer Funktionalität oder einfach nach einer neueren Version desselben Programms suchen.
Bei welchen Anwendungen dabei nach welcher Art von Passwort gefragt werden soll, können Sie als Administrator des UNIX-Systems konfigurieren. Dabei haben Sie auf den Client-Rechnern, von denen Anfragen wie telnet gestartet werden, keinerlei Software zu installieren. Sie haben lediglich auf den UNIX-Hosts einzugreifen, deren Passwortüberprüfungsmethoden Sie in Ihrem lokalen Netz optimieren wollen.
Wir unterstützen zur Zeit UNIX Versionen, die PAM (Pluggable Authentication Modules) verwenden. Auf der CD befinden sich Binärdateien für folgende Betriebssystemversionen:

- SuSE Linux bzw. openSUSE ab Version 8.3
- Solaris ab Version 9

Zur Installation des SecOVID-Einmalpasswortschutzes in Ihrem lokalen UNIX-Netz loggen Sie sich an den UNIX-Hosts, deren Passwortüberprüfungsmethoden Sie verbessern wollen, als root ein und führen Sie jeweils folgende Schritte durch:

- *Installation des PAM-SecOVID-Authentifikationsmoduls:*
  - Kopieren Sie die Datei `$UNIX/clients/pam/pam_ovid_auth.so` nach `/usr/lib/security/` (Solaris) bzw. `/lib/security/` (Linux).
  - Kopieren Sie die Datei `/etc/SecOvid/ovidcomm.pub` vom SecOVID-Server (dieser erzeugt die Datei, wenn er zum ersten Mal gestartet wird) auf alle Rechner, deren PAM-Module Sie modifizieren wollen. Speichern Sie die Kopie dabei unter demselben Pfad und Namen, also als `/etc/SecOvid/ovidcomm.pub`
  - Editieren Sie `/etc/SecOvid/ovid_config` auf allen betroffenen Rechnern. In dieser Datei müssen Sie die IP-Adresse des SecOVID-Servers und die Portnummer eintragen, auf der dieser RSA-Anfragen entgegennimmt (vgl. den Eintrag hinter `secovid` in `/etc/services` auf dem Rechner des SecOVID-Servers). IP-Adresse und Portnummer sind durch Doppelpunkt zu trennen. Auf Windows-Systemen muss die Datei mit einer leeren Zeile (newline) enden. Die Datei könnte z.B. folgenden Inhalt haben: `192.168.88.181:1647`
- *Festlegen der Sicherheits-Policy*
  Schauen Sie sich die Beispielkonfigurationen im Verzeichnis `$UNIX/clients/pam` auf der CD an und modifizieren Sie Ihre lokale PAM Konfiguration entsprechend (s.u.). Beachten Sie insbesondere, dass es möglich ist, mehrere Passwortmechanismen nacheinander zu verwenden.

- *Konfiguration unter SuSE Linux*
  Die Konfigurationsdatei, die auf Ihrem Linux-System die Sicherheitspolicy für eine bestimmte Anwendung festlegt, trägt im allgemeinen den Namen der betreffenden Client-Anwendung und liegt im Verzeichnis /etc/pam.d. Beachten Sie jedoch, dass z.B. die Konfigurationsdatei login für mehrere Anwendungen maßgeblich ist, nämlich lokales Konsolen-Login, telnet und teilweise rlogin, welches je nach Konfiguration nach einem ersten fehlgeschlagenen Anmeldungsversuch eine Verbindung mit dem Standard-Login herstellt. (Konsultieren Sie hierzu auch die Dokumentation Ihres Betriebssystems unter dem Stichwort „PAM“. Eine gute Zusammenfassung ist auch unter http://www.redhat.com/linux-info/pam/ erhältlich.)

Nehmen wir an, künftig soll ein rlogin auf einen Linux-Rechner nur noch unter Angabe eines gültigen Einmalpasswortes möglich sein. Dann editieren Sie die Datei `/etc/pam.d/rlogin` entsprechend Ihres Entschlusses: Falls Benutzer beim rlogin statt eines UNIX-Passwortes ein Einmalpasswort vorlegen sollen, so ersetzen Sie die Zeile, in der festgelegt wird, dass ein UNIX-Passwort eingegeben werden muss, durch eine entsprechende SecOVID-Zeile.
Ersetzen Sie also in der Datei `/etc/pam.d/rlogin` die Zeile:
`auth required /lib/security/pam_unix.so`
durch die Zeile:
`auth required /lib/security/pam_ovid_auth.so`.
Auf der CD finden Sie in der Datei `$UNIX/clients/pam/rlogin` folgendes Konfigurationsbeispiel:

```
# %PAM-1.0
auth     required    /lib/security/pam_securetty.so
auth     sufficient  /lib/security/pam_rhosts_auth.so
auth     requisite   /lib/security/pam_unix.so #set_secrpc
auth     required    /lib/security/pam_nologin.so
auth     required    /lib/security/pam_mail.so
auth     required    /lib/security/pam_ovid_auth.so
account  required    /lib/security/pam_unix.so
password required    /lib/security/pam_unix.so
session  required    /lib/security/pam_unix.so none # debug or trace
```

Nehmen wir an, dies wäre der Inhalt Ihrer Datei `/etc/pam.d/rlogin`. Bei einer rlogin-Anfrage auf den Rechner findet der rlogin-Dämon in dieser Datei, wie er Benutzer authentifizieren soll. Für die Authentifikation sind nur die Zeilen relevant, die mit `auth` beginnen. Entsprechende Zeilen werden der Reihe nach abgearbeitet und logisch durch „UND“ verknüpft. Das bedeutet in unserem Beispiel:
Ausreichend (die Zeile mit „`sufficient`“) für eine erfolgreiche Authentifikation ist ein positiver Entscheid durch das Authentifikationsmodul `/lib/security/pam_rhosts_auth.so`. (Dieses prüft, ob der Benutzer ohne Überprüfung eines Einmalpasswortes auf Grund der Adresse seines anfragenden Rechners vertrauenswürdig ist.) Falls dieses Modul nicht zu einem positiven Entscheid kommt, müssen alle anderen PAM-Authentifikationmodule, die in unserem Beispiel mit `auth required` oder `auth requisite` beginnen, zu einem positiven Entscheid kommen, damit der anfragende User eingeloggt wird. Insbesondere muss der anfragende Benutzer also ein korrektes UNIX-Passwort (PAM-Modul `/lib/security/pam_unix.so`) und ein korrektes Einmalpasswort (PAM-Modul `/lib/security/pam_ovid_auth.so`) vorlegen können. Kommt im Falle `auth requisite` das entsprechende Authentifikationsmodul zu einem negativen Ergebnis, so wird der Benutzer sofort abgelehnt, und die ggf. noch folgenden Zeilen in der Konfigurationsdatei werden für die Authentifikation nicht mehr abgearbeitet. Hingegen werden nach dem negativen Entscheid eines Authentifikationsmoduls im Falle einer Zeile, die mit `auth required` beginnt, die folgenden Zeilen abgearbeitet, obwohl schon feststeht, dass der Benutzer abgelehnt wird. (Ein Angreifer erhält in diesem Falle aber keine Auskunft darüber, welche Passwortmethode bei der Überprüfung zu einem negativen Ergebnis gekommen ist.) Beachten Sie insbesondere, dass es eine relativ einfache „Denial of Service“-Attacke für den Einmalpasswortmechanismus gibt: Geben Sie einem solchen Loginservice so lange falsche Passwörter, bis der attackierte User gesperrt ist. Daher schlagen wir die Überprüfung des statischen Passworts mit der Bedingung `auth requisite` vor, so dass nur ein User, der das statische Passwort kennt, überhaupt ein Einmalpasswort eingeben kann.
Das bedeutet, dass ein Angreifer, der eine „Denial of Service“-Attacke starten kann, bereits vollen Zugang zu

einem nur durch Standardmechanismen geschützten Netz hätte.
Detaillierte Informationen zur Bedeutung der Einträge finden Sie in der Dokumentation Ihres Betriebssystems unter dem Stichwort „PAM“.
Beachten Sie bitte, dass Sie in der Konfigurationsdatei `/etc/pam.d/other` das Verhalten von Anwendungen festlegen können, zu denen im Verzeichnis `/etc/pam.d/` keine explizite Konfigurationsdatei existiert. Ist hier eine Ablehnung und eine Meldung, welcher Service die Ablehnung verursachte, einprogrammiert, so kann man mit diesem Feature insbesondere herausfinden, wie der genaue Name einer Konfigurationsdatei lauten sollte. Dieser ist in der Praxis gelegentlich nicht so klar ersichtlich, wie er das eigentlich sein sollte.
Zum Einrichten des SecOVID-Einmalpasswortschutzes in anderen Anwendungen editieren Sie in analoger Weise die für die Anwendung relevante Konfigurationsdatei. Beispielsweise editieren Sie die Datei `/etc/pam.d/login` für die Anwendungen `login`, `telnet` und `rlogin` (selbiges wird, wenn der erste Authentifikationsversuch fehlschlug, häufig an ein Standard-Login übergeben, für das eben nicht mehr die Konfigurationsdatei `rlogin` zuständig ist).

Konfiguration unter Solaris In der PAM-Konfigurationsdatei `/etc/pam.conf` auf Ihrem Solaris-System wird das Authentifikationsverhalten einer bestimmten Anwendung im allgemeinen in den Zeilen definiert, die mit dem Namen der entsprechenden Client-Anwendung beginnt. Beachten Sie jedoch, dass die Zeilen, die mit login beginnen, für mehrere Anwendungen maßgeblich sind, nämlich lokales Konsolen-Login, telnet und teilweise rlogin, welches je nach Konfiguration nach einem ersten fehlgeschlagenen Anmeldungsversuch eine Verbindung mit dem Standard-Login herstellt. (Konsultieren Sie hierzu auch die Dokumentation Ihres Betriebssystems unter dem Stichwort „PAM“ (z.B. liefert man pam.conf hilfreiche Informationen), eine gute Zusammenfassung ist auch von http://www.redhat.com/linux-info/pam/ erhältlich.)
Nehmen wir an, künftig soll ein rlogin auf einen Solaris-Rechner nur noch unter Angabe eines gültigen Einmalpasswortes möglich sein. Dann editieren Sie die Datei `/etc/pam.conf` entsprechend Ihres Entschlusses:
Falls Benutzer beim rlogin statt eines UNIX-Passwortes ein Einmalpasswort vorlegen sollen, so ersetzen Sie die Zeile, in der festgelegt wird, dass ein UNIX-Passwort eingegeben werden muss, durch eine entsprechende SecOVID-Zeile. Ersetzen Sie also in der Datei `/etc/pam.conf` die Zeile:
`rlogin auth required /lib/security/pam_unix.so.1`
durch die Zeile:
`rlogin auth required /lib/security/pam_ovid_auth.so`.
Auf der CD finden Sie in der Datei `$UNIX/clients/pam/pam.conf` folgendes Konfigurationsbeispiel:

```
#ident   "@(#)pam.conf 1.19              95/11/30 SMI"
# PAM configuration
#
# Authentication management
#
login    auth     required    /usr/lib/security/pam_unix.so.1
login    auth     required    /usr/lib/security/pam_dial_auth.so.1
rlogin   auth     sufficient  /usr/lib/security/pam_rhosts_auth.so.1
rlogin   auth     requisite   /usr/lib/security/pam_unix.so.1
rlogin   auth     required    /usr/lib/security/pam_ovid_auth.so
dtlogin  auth     required    /usr/lib/security/pam_unix.so.1
rsh      auth     required    /usr/lib/security/pam_rhosts_auth.so.1
other    auth     required    /usr/lib/security/pam_unix.so.1
# Account management
login    account  requisite   /usr/lib/security/pam_unix.so.1
dtlogin  account  required    /usr/lib/security/pam_unix.so.1
other    account  required    /usr/lib/security/pam_unix.so.1
# Session management
other    session  required    /usr/lib/security/pam_unix.so.1
# Password management
other    password required    /usr/lib/security/pam_unix.so.1
```

Nehmen wir an, dies wäre der Inhalt Ihrer Datei `/etc/pam.conf`. Bei einer rlogin-Anfrage auf den Rechner findet der rlogin-Dämon in dieser Datei, wie er Benutzer authentifizieren soll. Für die Authentifikation beim rlogin sind nur die Zeilen relevant, die mit `rlogin auth` beginnen. Entsprechende Zeilen werden der Reihe nach abgearbeitet und logisch durch „UND“ verknüpft. Das bedeutet in unserem Beispiel:
Ausreichend (die Zeile mit „`sufficient`“) für eine erfolgreiche Authentifikation ist ein positiver Entscheid durch das Authentifikationsmodul `/lib/security/pam_rhosts_auth.so.1`. (Dieses prüft, ob der Benutzer ohne Überprüfung eines Einmalpasswortes auf Grund der Adresse seines anfragenden Rechners vertrauenswürdig ist.) Falls dieses Modul nicht zu einem positiven Entscheid kommt, müssen alle anderen
PAM-Authentifikationmodule, die in unserem Beispiel mit
`rlogin auth required` oder `rlogin auth requisite` beginnen, zu einem positiven Entscheid kommen, damit der anfragende User eingeloggt wird. Insbesondere muss der anfragende Benutzer also ein korrektes UNIX-Passwort
(PAM-Modul `/lib/security/pam_unix.so.1`) und ein korrektes Einmalpasswort
(PAM-Modul `/lib/security/pam_ovid_auth.so`) vorlegen können. Kommt im Falle `rlogin auth requisite` das Authentifikationsmodul zu einem negativen Ergebnis, so wird der Benutzer sofort abgelehnt, und die ggf. noch folgenden Zeilen in der Konfigurationsdatei für rlogin werden für die Authentifikation nicht mehr abgearbeitet. Hingegen werden nach dem negativen Entscheid eines Authentifikationsmoduls im Falle einer Zeile, die mit `rlogin auth required` beginnt, die folgenden Zeilen abgearbeitet, obwohl schon feststeht, dass der Benutzer abgelehnt wird. (Ein Angreifer erhält in diesem Falle aber keine Auskunft darüber, welche Passwortmethode bei der Überprüfung zu einem negativen Ergebnis gekommen ist.) Beachten Sie insbesondere, dass es eine relativ einfache „Denial of Service“-Attacke für den Einmalpasswortmechanismus gibt: Geben Sie einem solchen Loginservice so lange falsche Passwörter, bis der attackierte User gesperrt ist. Daher schlagen wir die Überprüfung des statischen Passworts mit der Bedingung `auth requisite` vor, so dass nur ein User, der das statische Passwort kennt, überhaupt ein Einmalpasswort eingeben kann. Das bedeutet, dass ein Angreifer, der eine „Denial of Service“-Attacke starten kann, bereits vollen Zugang zu einem nur durch Standardmechanismen geschützten Netz hätte.
Detaillierte Informationen zur Bedeutung der Einträge finden Sie in der Dokumentation Ihres Betriebssystems unter dem Stichwort „PAM“.
Beachten Sie bitte, dass Sie über den Eintrag `other` das Verhalten von Anwendungen festlegen können, die in der Konfigurationsdatei nicht explizit aufgeführt sind. Ist hier eine Ablehnung und eine Meldung, welcher Service die Ablehnung verursachte, einprogrammiert, so kann man mit diesem Feature insbesondere herausfinden, wie der genaue Name einer Konfigurationsdatei lauten sollte. Dieser ist in der Praxis gelegentlich nicht so klar ersichtlich, wie er das eigentlich sein sollte.
Zum Einrichten des SecOVID-Einmalpasswortschutzes in anderen Anwendungen editieren Sie in analoger Weise die für die Anwendung relevanten Zeilen in der Datei `/etc/pam.conf`. Beispielsweise editieren Sie die Zeilen, die mit `login auth` beginnen, für die Anwendungen`login`, `telnet` und `rlogin` (selbiges wird, wenn der erste Authentifikationsversuch fehlschlug, häufig an ein Standard-Login übergeben, für das eben nicht mehr die Konfigurationsdatei `rlogin` zuständig ist).

Verwendung von ssh Beachten Sie, dass die am weitesten verbreitete Version von ssh, nämlich OpenSSH, zunächst jeweils speziell für das Betriebssystem OpenBSD und ohne PAM-Unterstützung entwickelt wird. Dadurch ergeben sich einige spezifische Probleme bei der Verwendung unseres PAM-Moduls in Kombination mit OpenSSH.

- In einigen OpenSSH-Versionen funktioniert die PAM-Unterstützung überhaupt nicht korrekt.
- Es gibt immer wieder PAM-spezifische Kompatibilitätsprobleme zwischen SSH-Client und SSH-Server, wenn unterschiedliche SSH-Versionen verwendet werden. Sollten Sie trotz korrektem Password abgelehnt werden, können Sie versuchen, auf ClientSeite explizit verschiedene Protokollversionen zu probieren (z.B. mittels der Optionen `-1` oder `-2`), meistens funktioniert zumindest eine Version doch.
- In OpenSSH-3.7.1p2 und evtl. auch in neueren Versionen ist die PAM-Unterstützung serverseitig defaultmäßig ausgeschaltet und muss explizit durch Hinzufügen der Zeile `UsePAM yes` zur Datei `sshd_config` eingeschaltet werden.

2.3.10 Installation eines RSA-Clients*

Falls Sie den SecOVID-Einmalpasswortmechanismus in irgendeinem Produkt anwenden wollen, welches über eine RSA-Schnittstelle verfügt, so installieren Sie das betreffende Produkt und tragen Sie an der vorgesehenen Stelle die IP-Adresse, Portnummer und Public Key des SecOVID-Servers ein. In der Regel haben Sie dazu die Dateien `ovid_config` und `ovidcomm.pub` zu editieren (siehe Abschnitt 2.3.6.

2.4 Das GetPWD Programm

Der SecOVID Reader III erzeugt OTPs offline, d.h. ohne Anschluss an den PC. Dazu müssen die OTPs entsprechend vom Display des Readers abgelesen und manuell in die Bildschirmmaske eingegeben werden. Um diesen Vorganz etwas komfortabler zu gestalten, gibt es ein kleines Programm, das es dem Benutzer ermöglicht, ein Passwort mit Hilfe eines beliebigen an den Rechner angeschlossenen KOBIL Chipkartenterminals zu berechnen. Nachdem man mit diesem Programm ein Passwort berechnet hat, kann es einfach mit Mausklick in die gewünschte Eingabemaske eingefügt werden. Auf diese Weise kann sich der Benutzer das manuelle Abtippen des Passwortes vom Display des Taschenkartenlesers ersparen.
Für den Betrieb mit KOBIL mIDentity ist dieses Programm nicht notwendig, da die mIDentity Software es bereits enthält und mit Hilfe der Simple-Sign-On Funktion sogar das Ausfüllen der Anmelde-Dialogboxen vollständig automatisierbar ist.
Um das GetPWD Programm zu nutzen, gehen Sie bitte vor wie folgt:

1. **Installation des KOBIL-Chipkartenterminals (z.B. SecOVID Reader III)**
   - Folgen Sie den Instruktionen zur Installation, die dem Chipkartenterminal beiliegen.
2. **Installation** `getpwd`
   - **Unter Windows**
     Starten Sie das Programm `Setup.exe` im Verzeichnis `\win32\clients\getpwd`. Folgen Sie den Anweisungen des Installationsprogramms.
   - **Unter Linux oder Solaris**
     Starten Sie das Programm `$UNIX/install_getpwd.sh` auf der SecOVID-CD. Wählen Sie die gewünschte Sprachversion (deutsch oder englisch) aus. Installieren Sie außerdem die Shared Objects für GTK+-1.2.10 in einem Verzeichnis Ihrer Wahl und nehmen Sie dieses Verzeichnis global in den `LD_LIBRARY_PATH` auf.

Kapitel 3

Initiale Konfiguration und Bedienung im laufenden Betrieb

3.1 SecOVID-Serversystem

3.1.1 Konfiguration des SecOVID-Servers

Die Konfigurationsdateien und Testprogramme für den SecOVID-Server finden Sie unter `/etc/SecOvid/` :

- `/etc/SecOvid/ovid_licence` enthält die Lizenz, die vom SecOVID-Server beim Starten überprüft wird. Diese wurde Ihnen per Diskette oder email zugesandt.
- `/etc/SecOvid/db_allow` enthält einerseits die IP-Adressen der Rechner, von denen auf die SecOVID-Datenbank zugegriffen werden darf, andererseits die IP-Adressen der Rechner, auf denen die SecOVID-Datenbank gespiegelt werden soll. Die beiden Schlüsselwörter `allow` und `mirror` am Zeilenanfang zeigen an, ob von den folgenden IP-Adressen aus auf die Datenbank zugegriffen werden darf, oder ob die Datenbank auf diese Rechner gespiegelt werden soll. Verschiedene IP-Adressen werden dabei jeweils durch Leerzeichen getrennt. Aus Gründen der Kompatibilität zu früheren Versionen werden Zeilen, die mit keinem dieser beiden Schlüsselwörter anfangen, so behandelt, als stünde am Zeilenanfang ein `allow`. Damit könnte eine solche Datei z.B. wie folgt aussehen:

  ```
  allow 127.0.0.1
  mirror 192.168.1.4 192.168.1.5
  allow 192.168.1.113 192.168.1.114
  192.168.1.115
  ```

- `/etc/SecOvid/clients` enthält die IP-Adressen und symmetrischen Geheimnisse (shared secrets) der RADIUS-Clients (bzw. RADIUS-Server, die eine Anfrage an den vorliegenden Server stellen), die mit dem SecOVID-Server kommunizieren dürfen. Das betreffende Geheimnis müssen Sie jeweils auch in der entsprechenden Konfigurationsdatei auf dem Rechner des RADIUS-Clients eintragen. Die Geheimnisse (bis zu 15 alphanumerische Zeichen) werden dazu verwendet, die gesamte Kommunikation zwischen SecOVID-Server und RADIUS-Client zu verschlüsseln. Kommentaren stellen Sie bitte das Zeichen ‚#' voran. Auf Grund des sicherheitssensitiven Inhalts der Datei sollte nur root Lese- und Schreibzugriff auf die Datei haben. Die Datei könnte z.B. folgenden Inhalt haben:

  ```
  #--------------------------
  #localhost skjdageregz
  192.168.88.167 kdj-sfg.kl3jdfg
  192.168.88.161 kldfgj12?WM!9cf
  #--------------------------
  ```

- `/etc/SecOvid/ovid_ping` ist ein einfaches RSA-Client-Programm, mit dem Benutzername und Einmalpasswort testweise an den SecOVID-Server geschickt werden können. Bitte achten Sie darauf, dass auf dem Rechner, auf dem Sie `ovid_ping` starten, die Dateien `/etc/SecOvid/ovidcomm.pub` und `/etc/SecOvid/ovid_config` existieren. Informationen über den notwendigen Inhalt der beiden Dateien finden Sie in Abschnitt 2.1.7. Testen Sie mit diesem Programm, ob falsche Einmalpasswörter vom SecOVID-Server abgelehnt und korrekte akzeptiert werden:

  `/etc/SecOvid/ovid_ping -u username/password`, z.B.
  `/etc/SecOvid/ovid_ping -u testuser/98423623`

- `/etc/SecOvid/radping` ist ein einfaches RADIUS-Client-Programm, mit dem Benutzername und Einmalpasswort testweise über das RADIUS-Interface an den SecOVID-Server geschickt werden können. Testen Sie mit diesem Programm, ob falsche Einmalpasswörter vom SecOVID-Server abgelehnt und korrekte akzeptiert werden:

  - `radping -u username/password -S SecovidServerIP[:serverPort] -K secretKey`, z.B.
  - `radping -u smith/26011735 -S 192.168.1.1 -K secret`

- `/etc/SecOvid/tacping` ist ein einfaches TACACS+-Client-Programm, mit dem Benutzername und Einmalpasswort testweise über das TACACS+-Interface an den SecOVID-Server geschickt werden können. Testen Sie mit diesem Programm, ob falsche Einmalpasswörter vom SecOVID-Server abgelehnt und korrekte akzeptiert werden:

  - `tacping -u username/password -S SecovidServerIP -K secretKey`, z.B.
  - `tacping -u smith/26011735 -S 192.168.1.1 -K secret`

Folgende Dateien werden zudem vom SecOVID-Server beim Start erzeugt, falls sie nicht bereits existieren:

- `/etc/SecOvid/ovid_data` die SecOVID-Datenbasis mit den Benutzereinträgen (und ggf. Administratoreinträgen)
- `/etc/SecOvid/ovidcomm.key` der private Schlüssel des SecOVID-Servers für die gesicherte Kommunikation mit den RSA-Clients (z.B. UNIX-PAM-Clients). Dieser wird vom SecOVID-Server für Entschlüsselung und Signatur von Passwortanfrage und Antwort verwendet. Der SecOVID-Server erzeugt das Paar aus privatem und öffentlichem Schlüssel bei seinem ersten Start und legt die betreffenden Dateien an. Somit findet die Schlüsselerzeugung bei Ihnen vor Ort statt.
- `/etc/SecOvid/ovidcomm.pub` der öffentliche Schlüssel des SecOVID-Servers für die gesicherte Kommunikation mit den SecOVID-RSA-Clients. Dieser sollte hier als Backup liegen und muss auf allen SecOVID-RSA-Clientrechnern im vorgesehenen Pfad gespeichert sein.
- `/etc/SecOvid/ovid.proto` das Log-Datei des SecOVID-Servers. Hier werden alle Passwortanfragen mit den Ergebnissen der Überprüfung protokolliert.
- `/etc/SecOvid/secovid` der eigentliche SecOVID-Dämon.
- `/etc/SecOvid/secovid.sh` Skript, welches den SecOVID-Dämon startet.

Serverfunktionalität des SecOVID-Servers Der SecOVID-Server ist ein Authentifizierungs-Server, der Einmalpasswortanfragen prüfen kann und über eine RADIUS-Schnittstelle (RADIUS Version 2), eine TACACS+-Schnittstelle (TACACS+ Protocol v 1.78 konform)und eine RSA-Schnittstelle verfügt.
Im Vergleich zu einem RADIUS-,TACACS+-Server verfügt der SecOVID-Server über ein eigenes Accounting und kann statische Passwörter überprüfen, Benutzerprofile speichern und alle möglichen RADIUS-,TACACS+-Attribute wie ein beliebiger RADIUS-, TACACS+-Server handhaben. Bzgl. der RADIUS-Attribute erfolgt die Administration des SecOVID-Servers genau wie bei klassischen RADIUS-Servern. Der einzige Unterschied besteht darin, dass sich die von der Verwaltung von RADIUS-Servern bekannten Dateien aus dem Verzeichnis

`/etc/raddb/` beim SecOVID-Server unter `/etc/SecOvid/` finden. Bzgl. der TACACS+-Attribute erfolgt die Konfiguration des SecOVID-Servers genau wie bei TACACS+-Servern.

Folgende Dateien können zur Konfiguration verwendet werden:

- `/etc/SecOvid/users` die von RADIUS-Servern bekannte Userdatenbank zum Abspeichern von Benutzerprofilen (Authentifikationsdaten und Benutzerrechte). Beachten Sie, dass Sie zumindest eine minimale Version dieser Datei benötigen, die die Zeile

  ```
  DEFAULT     Password="OVID"
  ```

  beinhaltet.

  WARNUNG: Mit einer leeren oder ganz ohne eine Userdatenbank `/etc/SecOvid/users` werden alle RADIUS-Anfragen an den SecOVID-Server abgelehnt.

- `/etc/SecOvid/dictionary` die von RADIUS-Servern bekannte Datei, die alle möglichen RADIUS-Attribute aufführt. Sie ist dafür verantwortlich, dass die verständlichen Attributnamen aus der Konfigurationsdatei `/etc/SecOvid/users` in die entprechenden Nummern des RADIUS-Protokolls übersetzt werden. Dabei können Sie die Attributnamen nach Belieben ändern, um etwa eine Anpassung an eine bereits existierende `users`-Datei vorzunehmen. Zu beachten ist nur, dass auch wirklich alle Attributnamen der `users`-Datei definiert werden. Insbesondere können zusätzlich Hersteller-spezifische Attribute nach dem standardisierten Mechanismus ergänzt werden.

- `/etc/SecOvid/tacacs` Konfigurationsdatei zum Abspeichern von Benutzerprofilen (Authentifikationsdaten und Benutzerrechte).

  Beachten Sie, dass Sie zumindest eine minimale Version dieser Datei benötigen, die die Zeile

  ```
  key = "secret key"
  default authentication = onetime
  ```

  beinhaltet.

  WARNUNG: Mit einer leeren oder ganz ohne eine Userdatenbank `/etc/SecOvid/tacacs` werden alle TACACS+-Anfragen an den SecOVID-Server abgelehnt.

Detaillierte Informationen zum Verwenden von RADIUS-Attributen und dem Einrichten von Userprofilen finden Sie im beiliegenden Dokument How to configure user information. Dieses Dokument liegt zur Zeit ausschließlich in Englisch vor.

Ändern der Portnummern des SecOVID-Servers In der Datei `/etc/services` (Unix) bzw. `c:\WINNT\system32\drivers\etc\services` (win32) werden die verschiedenen Interfaces des SecOVID-Dämons zu Portnummern zugeordnet. An dieser Stelle sollten Sie sicherstellen, dass die von SecOVID benutzten Portnummern von keinem anderen Dämon verwendet werden. Die Datei `/etc/services` bzw. `c:\WINNT\system32\drivers\etc\services` (win32) enthält folgende für SecOVID relevante Defaulteinträge:

```
tacacs          49/tcp
tacacs          49/udp
radius          1812/udp
secovid         1647/udp
secoviddb       1113/udp
```

Bemerkung: `radius` wird unter Windows standardmässig der Port 1812 zugewiesen.

Der SecOVID-Dämon wartet nun an dem Port namens `radius` (im vorliegenden Fall also unter Portnummer `1812`) auf RADIUS-konforme Passwortanfragen und am `secovid`-Port (Portnummer `1647`) auf RSA-verschlüsselte Passwortanfragen. Das Administrationstool greift über den `secoviddb`-Port (Port `1113`) auf die im Arbeitsspeicher residierende Datenbasis zu.

Starten und Stoppen des SecOVID-Servers unter Unix-Systemen

- **Starten**
  Sie starten den SecOVID-Dämon durch Eingabe des Kommandos
  `/etc/rc.d/rc2.d/S65secovid start` (Linux) bzw. `/etc/rc2.d/S65secovid start` (Solaris)
  (oder `/etc/SecOvid/secovid.sh`).

- **Stoppen**
  Sie stoppen den SecOVID-Dämon durch Eingabe des Kommandos
  `/etc/rc.d/rc2.d/S65secovid stop` (Linux) bzw. `/etc/rc2.d/S65secovid stop` (Solaris).

  Alternativ können Sie auch die Prozess-ID des SecOVID-Dämons bestimmen und dann den Prozess mittels `kill <Prozess-ID>` stoppen, z.B. ergibt sich unter Linux folgender Ablauf:

```
# ps x | grep secovid
 228    ?    S     0:00    /etc/SecOvid/secovid
# kill 228
```

  während dieselbe Operation unter Solaris wie folgt aussehen könnte:

```
# ps -e | grep secovid
   228 ?        0:00 secovid
# kill 228
```

Starten und Stoppen des SecOVID-Servers unter WinNT

- Der SecOVID-Server kann unter Systemsteuerung > Verwaltung > Dienste gestartet bzw. gestoppt werden. Wahlweise können in einer „Eingabeaufforderung“ auch die Kommandos `net start secovid` bzw. `net stop secovid` benutzt werden.

Editieren der SecOVID-Datenbank Jegliches Editieren und Anschauen der SecOVID-Datenbank wird über das SecOVID-Administrationstool vorgenommen. Zum Starten des Administrationstools muss der SecOVID-Dämon `/etc/SecOvid/secovid` (Unix) bzw. der SecOVID-Service (Win32) laufen, denn dieser enthält den Datenbankserver.

Ändern der SecOVID-Server Startoptionen Zum Ändern der Startoptionen editieren sie auf UNIX-Systemen das Startskript `secovid.sh` indem sie die gewünschten Optionen einfach an das Startkommando `./secovid` anhängen. Auf Windows-Systemen tragen Sie die gewünschten Parameter in die Datei `secovid.arg` im Verzeichnis `\etc\SecOvid` ein (ggf. müssen Sie diese Datei zunächst erzeugen; beachten Sie, dass Sie je nach verwendetem Editor die Endung `.txt` entfernen müssen), wobei jede Option in eine eigene Zeile geschrieben werden sollte.
Auf UNIX-Systemen können Sie, wenn sie eine Option nur einmal verwenden wollen, auch den Service stoppen und mittels `/etc/SecOvid/secovid <Optionen>` die gewünschten Optionen direkt angeben.

Logging Durch die folgenden Parameter können Sie ein detaillierteres Logging des SecOVID Servers einschalten.

- Ändern des Namens des Logfiles (default „ovid.proto“):
  `-l FILE` oder
  `--log=FILE`

- Einschalten des RADIUS-Debug-Loggings:
  `-xr FILE` oder
  `--debug-radius=FILE`

- Einschalten des TACACS+-Debug-Loggings:
  `-xt FILE` oder
  `--debug-tacacs=FILE`

Thread-Management Der SecOVID–Server unterstützt auch die Anbindung an andere RADIUS-Server zur Authentifikation. Da sich hier je nach Art des fremden RADIUS-Servers lange Wartezeiten für die Verifikation eines Passworts ergeben können (bis zu mehrere Sekunden), verwendet der SecOVID–Server Multithreading, wobei je nach Bedarf bis zu einer beim Start festgelegten Obergrenze Threads erzeugt werden.
Diese Obergrenze können Sie mit dem Parameter
`-t num` oder
`--threads=num`, `0 < num < 1001`
konfigurieren.
Wenn Sie nur den SecOVID-Server zur Authentifikation verwenden, ist zur Maximierung des Durchsatzes die Option `-t 1` zu empfehlen, der Default–Wert ist `-t 100` um zu gewährleisten, dass auch dann noch eine ausreichende Performance gewährleistet ist, wenn die Passwörter vieler User an einen anderen, langsamereren Server zur Verifikation weitergeleitet werden.

Suchtiefe und Fehlergrenze für Einmalpasswörter Wie wir bereits in Abschnitt 1 bemerkten, akzeptiert der SecOVID-Server auch Einmalpasswörter, die ihm nicht aktuell erscheinen, aber in der Liste der künftigen Passwörter erscheinen werden. In der Standardeinstellung akzeptiert der SecOVID-Server die ersten zehn Einmalpasswörter. Wir wollen hier von der Suchtiefe des SecOVID-Servers sprechen. Um diese zu ändern können Sie den Parameter
`--depth=num` oder
`-d num`, `2 < num < 51`
verwenden.
Darüber hinaus können Sie für einzelne Benutzer, die so viele Passwörter ohne Verwendung generiert haben, dass sie aus der Suchtiefe des SecOVID-Servers rausfallen, über den Button „Resynchronisation“ im Administrationstool die Suchtiefe um einen Faktor 10 gegenüber der eingestellten Suchtiefe erhöhen. In diesem Fall muss der Benutzer sich aber nun zweimal mit unmittelbar aufeinander folgenden Einmalpasswörtern anmelden, wobei erst der zweite Anmeldeversuch zum Erfolg führt.
Eine weitere Konfigurationsmöglichkeit betrifft die Fehlergrenze, d.h. die Anzahl der falschen Passwörter, die ein Benutzer eingeben darf, bevor er gesperrt wird. Normalerweise wird der Benutzer nach dem 10-ten falschen Passwort gesperrt. Je nach Anwendung oder z.B. wenn sie statt 8-stelliger Passwörter 6-stellige verwenden, kann es sinnvoll sein, diese Grenze herab zu setzen. Dies können Sie mit dem Parameter
`--error-limit=num` oder
`-e num`, `2 < num < 11`
konfigurieren. Dann wird der Benutzer nach dem `num`-ten statt nach dem zehnten Fehlversuch gesperrt.

Groß- und Kleinschreibung von Benutzernamen Normalerweise unterscheidet der SecOVID-Server bei Benutzernamen auch nach Groß- und Kleinschreibung, d.h. die Benutzer `test`, `Test` und `TEST` sind 3 verschiedene Benutzer. Wenn sich Benutzer `test` irrtümlich als `Test` anzumelden versucht, wird die Passwortanfrage als falsch zurückgewiesen.
Dies ist insbesondere in einem Windows-Umfeld nicht immer erwünscht. Deshalb bietet der Parameter
`--ignore-case` oder
`-ic`
die Möglichkeit, dieses Verhalten zu ändern, d.h. wenn Sie beim Starten des SecOVID-Servers diesen Parameter angeben, werden die Benutzer `test`, `Test` und `TEST` als identisch behandelt, in der Datenbank wird allerdings stets die Schreibweise verwendet, die sie beim Anlegen des Benutzers verwendet haben. Beachten Sie dabei jedoch, dass Umlaute problematisch sind, d.h. wenn in einem Benutzernamen z.B. ein *ü* vorkommt, wird die Schreibweise mit *Ü* eventuell nicht korrekt erkannt (je nach Plattform auf der der Server bzw. der Client laufen). Anstelle der Verwendung dieser Option bietet es sich an, eine spezifische Policy für die Bildung von Benutzernamen zu verwenden, die überwacht, dass Benutzernamen nur in der gewünschten Schreibweise gebildet werden dürfen, z.B. nur aus Kleinbuchstaben, oder immer mit einem Großbuchstaben beginnend, der nur von Kleinbuchstaben gefolgt wird, oder wie auch immer es in Ihrer Umgebung angemessen ist.
Warnung: Falls Sie den Server zunächst ohne diese Option verwendet haben und Benutzer angelegt haben, deren Benutzernamen sich nur in der Groß- und Kleinschreibung unterscheiden (z.B. `test`, `Test` und `TEST`), dann werden jeweils alle diese Benutzer bis auf den (nach alphanumerischer Sortierung) letzten gelöscht, wenn Sie den SecOVID-Server mit dieser Option starten, eventuell vorhandene Datensätze für Chipkarten und Token gehen dabei verloren, so dass diese Chipkarten bzw. Token unbrauchbar werden. Wenn Sie eine solche Umstellung vornehmen wollen, empfiehlt es sich also, zunächst die Benutzernamen so zu ändern, dass keine „Duplikate“ auftreten werden und sodann ein Backup der Datenbank anzulegen, um gegebenenfalls eine Möglichkeit zu haben, übersehene Probleme zu korrigieren.

Bemerkung:
Denken Sie daran, dass Sie nach jedem Editieren des Files `secovid.sh` (Unix) bzw. `secovid.arg` (win32) den SecOVID Server neu starten müssen, damit die Änderungen übernommen werden.

Verwenden einer alten SecOVID-Datenbasis Wenn Sie bereits über eine SecOVID-Datenbasis verfügen und diese vom Administrationstool oder vom SecOVID-Server benutzt werden soll, so gehen Sie wie folgt vor. Beachten Sie dabei unbedingt die vorgegebene Reihenfolge:

- Stoppen Sie den SecOVID-Server.
- Sichern Sie die aktuelle SecOVID-Datenbasis durch Umkopieren der Datei `/etc/SecOvid/ovid_data`.
- Kopieren Sie die zu verwendende alte SecOVID-Datenbasis nach `/etc/SecOvid/ovid_data`. Beachten Sie, dass die Codierung der Datenbasis von der Architektur des verwendeten Rechners abhängt. Daher können Sie die SecOVID-Datenbasis eines PC i86 nicht ohne Weiteres auf eine SUN SPARC portieren.
- Starten Sie erneut den SecOVID-Server.

Warnung: Falls Sie die aktuelle SecOVID-Datenbasis umbenennen würden, den SecOVID-Server neu starten würden, anschließend Ihrer Datenbasis wieder den korrekten Namen `/etc/SecOvid/ovid_data` geben und nun über das Admintool einen Benutzer neu anlegen würden, so würde das Datei `/etc/SecOvid/ovid_data` überschrieben werden und Ihre Datenbasis wäre gelöscht. Versäumen Sie also niemals, den SecOVID-Server vor dem Umkopieren einer Datenbasis in `/etc/SecOvid/ovid_data` zu beenden und ihn anschließend erst zu starten.

Erzeugen eines neuen RSA-Schlüsselpaares So erzeugen Sie ein neues RSA-Schlüsselpaar für den SecOVID-Server:

- löschen Sie die Dateien, in denen die alten Schlüssel gespeichert waren (sofern diese überhaupt existieren): `/etc/SecOvid/ovidcomm.pub` und `/etc/SecOvid/ovidcomm.key`
- beenden Sie den SecOVID-Serverprozess
- starten Sie den SecOVID-Server neu: `/etc/SecOvid/secovid.sh`

Beachten Sie, dass Sie den neuen Public Key des SecOVID-Servers bei allen SecOVID-RSA-Clients (z.B. alle UNIX-Hosts, die durch SecOVID geschützt werden) bekannt machen müssen. Kopieren Sie dazu die Datei `/etc/SecOvid/ovidcomm.pub` an die vorgesehenen Stellen auf den RSA-Clientrechnern.

3.1.2 Bedienung des SecOVID-Administrationstools

Im folgenden beschreiben wir die Bedienung des SecOVID-Administrationstools („Admintool“) zur Benutzer- und Kartenverwaltung. Beachten Sie, dass Sie zur Bedienung des Administrationstools entweder ein registrierter Administrator (mit Administrator-Chipkarte) sein oder lokal auf derselben Maschine, auf der der SecOVID-Server läuft, arbeiten müssen.
Die SecOVID-Administratoren werden ebenfalls mit diesem Tool verwaltet. Daher müssen Sie bei einer ersten Installation zunächst das Administrationstool auf dem SecOVID-Server installieren und lokal (d.h. vom Rechner des SecOVID-Servers aus) SecOVID-Administratoren und Chipkarten für diese Administratoren erzeugen. Danach können Sie das SecOVID-Administrationstool auf weiteren Rechnern installieren und mit Hilfe der Administrator-Chipkarten remote auf den SecOVID-Server zugreifen.
Hinweis: Aus Sicherheitsgründen (starke Authentifikation des Administrators und Verschlüsselung aller übertragenen Daten) ist eine *Remote*-Administration des SecOVID-Servers ausschließlich per Administrator-Chipkarte möglich. Die *lokale* Administration des SecOVID-Servers ist sowohl mit als auch ohne Administratorchipkarte möglich. Wir empfehlen allerdings, die lokale Administration ohne Administrator-Chipkarte durch entsprechende Konfiguration (siehe unten) zu verbieten.

3.1.2.1 Erstmaliges Arbeiten mit dem Administrationstool

Wenn Sie zum ersten Mal mit dem SecOVID-Administrationstool arbeiten, so beachten Sie bitte folgendes:

- Installieren Sie das SecOVID-Administrationstool zunächst lokal auf dem Rechner des SecOVID-Server. (Zur Remote-Administration des SecOVID-Servers benötigen Sie eine SecOVID-Administratorchipkarte; diese kann anfangs nur lokal ausgestellt werden, siehe oben.)
- Konfigurieren Sie das Administrationstool so, dass Sie eine Verbindung zum SecOVID-Server aufnehmen können (Menüpunkt Programm > Konfiguration).
- *Anlegen von Einmalpasswort-Benutzern mit Chipkarte*
  Legen Sie einen Benutzer zunächst in der SecOVID-Datenbank an (Menüpunkt Datenbank > Neuer Benutzer). Schreiben Sie dann einen SecOVID-Einmalpasswortgenerator auf die für den Benutzer vorgesehene Chipkarte (Menüpunkt Chipkarten > Karte erzeugen). Details folgen weiter unten.
- *Anlegen von Einmalpasswort-Benutzern mit SecOVID Token*
  Importieren Sie die zu den SecOVID Tokens gehörende Datei mit den geheimen Tokendaten (Menüpunkt Token > Token importieren). In der Ansicht sehen Sie, dass die Seriennummern der Tokens nun auch als Benutzernamen dienen und die Textfelder für Nachname und Vorname leer sind. Benennen Sie nun die Benutzernamen um und tragen Sie Nachnamen und Vornamen der tatsächlichen Benutzer ein (rechter Mausklick auf den Benutzereintrag, „Information“ anklicken und die betreffenden Textfelder ändern). Eine detaillierte Beschreibung finden Sie weiter unten.

3.1.2.2 Starten des Administrationstools

Bei Starten des Programms muss bereits der SecOVID-Server in Betrieb sein. Zudem muss auf Nicht-Windows-Systemen die graphische Oberfläche X laufen.
Starten Sie das Administrationstool unter Unix durch die Kommandofolge

```
cd /etc/SecOvid
./SecOvid_admin.sh
```

bzw. unter Win32 durch

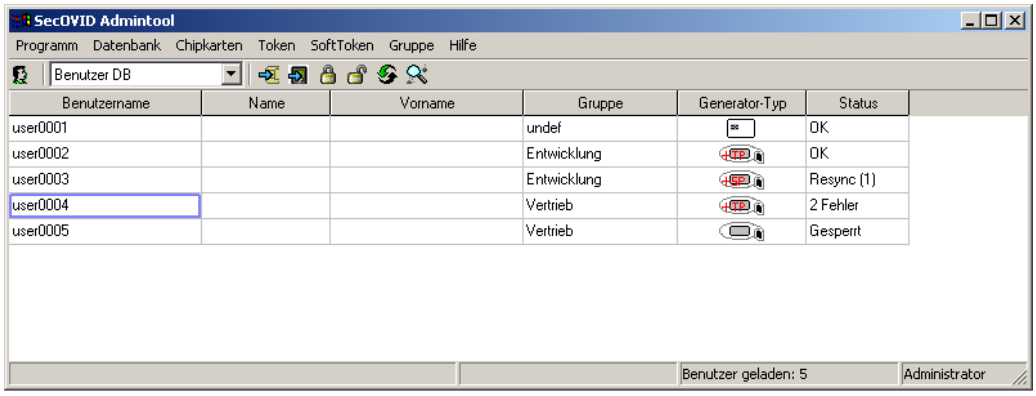
```
Win32 Programme > KOBIL Systems > SecOVID admintools > WxOVID
```

Sie erhalten nun ggf. die Aufforderung, Ihre Administrator-Chipkarte einzulegen und die PIN über die Tastatur des Chipkartenterminals einzugeben. Bestätigen Sie dies mit OK und geben Sie die PIN ein, sobald das Terminal diese anfordert. Dabei wird implizit vorausgesetzt, dass sich derselbe Administrator wie beim letzten Mal anmelden will.
Wenn dies nicht der Fall ist, oder wenn Sie keine Administrator-Chipkarte haben, oder wenn die eingegebene PIN falsch war, erscheint ein Dialog, in dem Sie „Abbrechen“ wählen können, um auch ohne Anmeldung mit dem Server kommunizieren zu können. Dies ist aus Sicherheitsgründen allerdings nur möglich, wenn das Admintool lokal auf demselben Rechner wie der SecOVID-Server läuft und wenn die Adresse `127.0.0.1` in der Datei `db_allow` eingetragen ist. Aus Sicherheitsgründen empfehlen wir, von dieser Möglichkeit ausschließlich zum Initialisieren der ersten Administrator-Chipkarte Gebrauch zu machen und danach die Adresse `127.0.0.1` aus der Datei `db_allow` zu löschen und den SecOVID-Server neu zu starten.
Normalerweise werden Ihnen nun alle im SecOVID-Server registrierten Benutzer angezeigt (u.U. sind noch keine Benutzer registriert).
Im negativen Fall erhalten Sie eine entsprechende Fehlermeldung - Überprüfen Sie die Einstellungen sowie die Installation des Chipkartenterminals. Stellen Sie beim Versuch einer Remote-Administration des SecOVID-Servers zudem sicher, dass Ihre Firewalls und sonstigen Paketfilter die Netzwerk-Kommunikation zwischen SecOVID-Admintool und SecOVID-Server auf dem Datenbankport (in der Regel Port 1113, siehe den Eintrag `secoviddb` in der Datei `/etc/services` des SecOVID-Servers) zulassen.

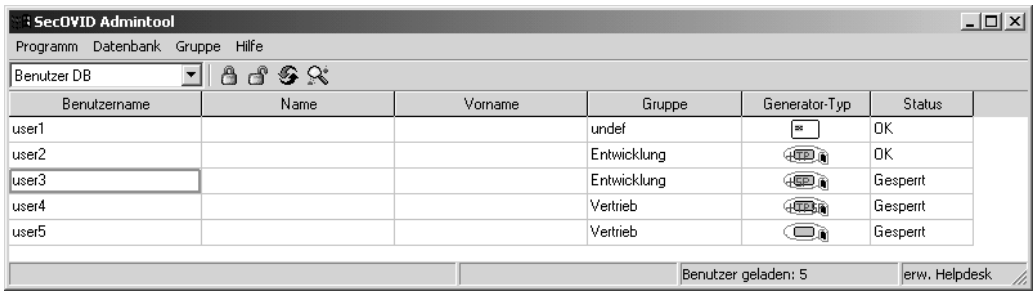
3.1.2.3 Die Ansicht

Nach erfolgreichem Start des Programms wird sofort die SecOVID-Datenbasis ausgelesen und angezeigt, die dem SecOVID-Server aktuell vorliegt. Sobald die Datenbank angezeigt wird, kann die Administrator-Chipkarte aus dem Chipkartenterminal entnommen werden, so dass dieses nun für das Beschreiben von Benutzer- oder Administratorchipkarten zur Verfügung steht.
Je nach Ihren Administratorenrechten sehen Sie folgende Ansicht:

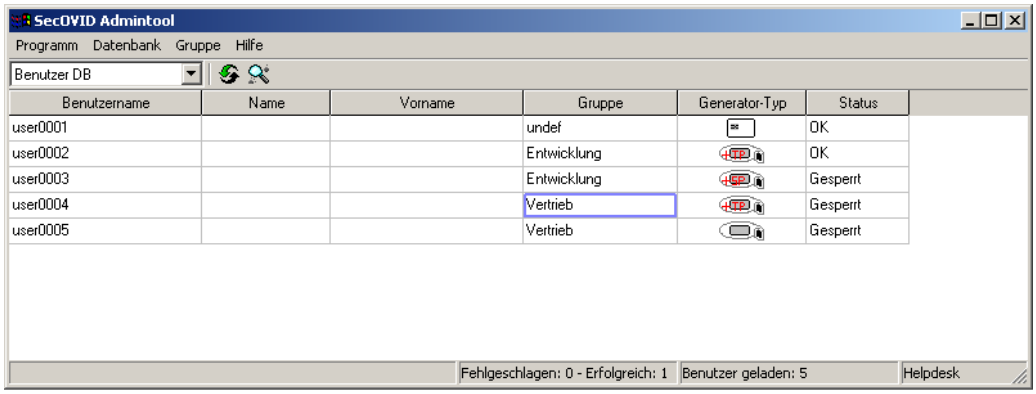
Administrator



Erweiterter Helpdesk



Helpdesk



Auf dem Bildschirm haben Sie nun eine Sicht auf Ihre Benutzer. (Die Sicht auf Ihre SecOVID-Administratoren erhalten Sie über das Menü „Datenbank“ > „Administratoren Datenbank“ oder über die Auswahlbox in der Buttonleiste.) Sie sehen zu jedem Eintrag:

- den Benutzernamen, Namen und Vornamen des Benutzers
- welcher Gruppe der Benutzer angehört
- welchen Passwortgenerator der Benutzer zum Erzeugen von SecOVID-Einmalpasswörter verwendet
- ob der Benutzer durch den Security-Administrator gesperrt wurde (also auch mit gültigen Einmalpasswörtern keinen Einlass ins System erhält)
- ob der Benutzer zur Resynchronisation vorgemerkt wurde („Resync (1)“), gerade das erste Passwort im Resynchronisationsprozess eingegeben hat („Resync(2)“), oder wie viele aufeinander folgende falsche Passwörter er seit dem letzten erfolgreichen Einloggen eingegeben hat (z.B. „2 Fehler“). „OK“ heißt: Das letzte vorgelegte Passwort war korrekt.

In der Statusleiste sehen Sie die Anzahl der geladenen Benutzer und den Administratoren-Typ des eingeloggten Administrator.
Auf die Menüleiste und die Buttonleiste gehen wir in den Punkten 3.1.2.4 und 3.1.2.5 ein.
Falls keine Verbindung zum SecOVID-Server hergestellt werden kann, etwa weil Ihre Administrator-Chipkarte eine andere ist als die des zuletzt tätigen Administrators oder weil der SecOVID-Server nicht läuft, so erhalten Sie eine entsprechende Warnung und das Fenster bleibt zunächst leer. In diesem Fall können Sie die Einstellungen über den Menüpunkt „Programm“ > „Konfiguration“ korrigieren.
Anlegen und Verwalten von SecOVID-Administratoren

SecOVID-Administratoren werden im wesentlichen wie normale Benutzer verwaltet, allerdings in einer eigenen Datenbank. Zwischen Benutzer- und Administratoren-Datenbank kann mittels eines eigenen Menüpunkts („Datenbank“ > „Administratoren Datenbank“) hin- und her geschaltet werden. Beachten Sie aber folgende Details:

1. Zum Anlegen einer Administrator-Chipkarte gehen Sie genauso vor, wie zum Anlegen eines Einmalpasswortgenerators auf einer Benutzer-Chipkarte. Beachten Sie allerdings, dass automatisch der Einmalpasswortgenerator Nr. 9 ausgewählt wird, wenn Sie eine Administrator-Chipkarte anlegen wollen.

2. Der Administratorname, mit dem Sie sich beim Admintool anmelden, ist der Benutzername (Administratorname) in der ersten Spalte in der Ansicht, die das Admintool bietet.

3. Die Namen für Administratoren und Benutzer werden getrennt verwaltet. Das heißt einerseits, dass kein Problem entsteht, wenn ein Administrator und ein Benutzer zufällig denselben Namen haben. Andererseits bedeutet das aber auch, dass ein Administrator, der auch den Einmalpasswort-Mechanismus benutzen will, in der Benutzerdatenbank nochmals angelegt werden muss, selbst wenn er dort den gleichen Benutzernamen verwendet. In diesem Fall muss zum einen der Menüpunkt „Karte erzeugen“ für den Administrator, zum anderen „Karte erzeugen“ für den Benutzer aktiviert werden, wobei im letzteren Fall für den Einmalpasswortgenerator eine andere Nummer als Nr. 9 ausgewählt werden muss; denn Nr. 9 wird bereits für den Administrator verwendet.

3.1.2.4 Die Menüleiste

Nacheinander werden nun die einzelnen Menüpunkte beschrieben, die einem Administrator zur Verfügung stehen. In Unterkapitel „Erweiterter Helpdesk“ bzw. „Helpdesk“ werden dann nur noch die Menüpunkte aufgezählt, die diese Admninstratoren-Typen verwenden können ohne näher auf die Punkte einzugehen.

Die Menüleiste als Administrator

Die Menüleiste eines Administrators enthält folgende Menüpunkte

Programm:

- ***Konfiguration:***
  Festlegen der Erreichbarkeit von SecOVID-Datenbank und Chipkartenterminal sowie Vorgabe der Defaultwerte, die im Programm benutzt werden. Dazu steht ein Dialog mit fünf Karteikarten zur Verfügung.
  Durch Anklicken des OK-Buttons wird geprüft, ob die eingestellten Werte sinnvoll sind. Falls die SecOVID-Datenbank oder das Chipkartenterminal nicht erreichbar sind, erhalten Sie eine entsprechende Warnung.

  – Die Seite „*Datenbank*“:
    * Unter „*Datenbank IP*“ tragen Sie die IP-Adresse oder den Namen des Computers ein, auf dem der SecOVID-Server läuft. Die sicherheitssensitive Datenbank wird so konfiguriert, dass nur von bestimmten Rechnern aus per Administrator-Chipkarte Remotezugriff möglich ist. (Die Datei `/etc/SecOvid/db_allow` enthält die IP-Adressen aller Rechner, die auf die SecOVID-Datenbank zugreifen dürfen.)
    * Unter „*Datenbank Port*“ tragen Sie die Port-Nummer ein, auf welcher der SecOVID-Server über sein Interface `secoviddb` (siehe `/etc/services`) auf Anfragen wartet (Standard-Wert: `1113`). Bitte beachten Sie: Für den SecOVID-Server ist die Port-Nummer maßgeblich, die in der Datei `/etc/services` hinter `secoviddb` vermerkt ist. Für das Admintool ist der Port maßgeblich, der im vorliegenden Textfeld eingetragen ist. Tragen Sie hier `0` ein, so sucht das Admintool auf dem vorliegenden Rechner nach dem Eintrag `secoviddb` in der Datei `/etc/services`. Falls ein solcher Eintrag existiert, so verwendet nun auch das Admintool die hier vermerkte Port-Nummer. Falls ein solcher Eintrag nicht existiert, erhalten Sie eine entsprechende Warnung.
    * Unter „*Pfad für Import/Export/Log*“ tragen Sie die Default-Pfade für die Import- und Exportdatei von SecOVID-Benutzerdaten sowie die Log-Datei ein. Die Pfade zur Import- oder Export-Datei können jeweils beim eigentlichen Importieren oder Exportieren nochmals angepasst werden. Bei jeder Import- oder Exportaktion wird die Datei `OvidLog.log` im hier gewählten Verzeichnis für die Log-Datei geschrieben. In dieser Datei werden eventuelle Fehler bei der Operation

protokolliert. Vermeiden Sie die Eingabe von `./` und `../` und bevorzugen Sie die Eingabe von absoluten Pfaden.

  * Unter „*Administratorname*“ geben Sie Ihren beim SecOVID-Server registrierten Administratornamen ein. Um lokal unverschlüsselt und ohne Chipkarte mit dem SecOVID-Server zu kommunizieren, geben Sie keinen Administratornamen an bzw. löschen Sie den Namen, der in diesem Feld steht.

– Die Seite „*Kartenterminal*“:

  * Tragen Sie unter „*COM Port*“ ein, an welcher COM-Schnittstelle Sie Ihr KOBIL-Chipkartenterminal angeschlossen haben. **Tip:** Probieren Sie im Zweifelsfall die verschiedenen Möglichkeiten aus, indem Sie jeweils den OK-Button anklicken. Haben Sie den falschen COM-Port eingestellt, so erscheint eine entsprechende Warnung (sofern Sie die entsprechende Checkbox aktiviert haben, s.u.).

  * Die Checkbox „*Beim Start automatisch suchen*“ können Sie aktivieren, wenn Sie wünschen, dass das Admintool beim Starten automatisch nach dem Kartenterminal sucht und das erste Kartenterminal benutzt, das gefunden wird. Aktivieren Sie diese Box nicht, so müssen Sie manuell den korrekten COM-Port eintragen. Dieser Wert wird beim Verlassen gespeichert. Dies kann etwa dann wünschenswert sein, wenn sie mehrere COM-Ports haben, an denen ein Gerät angeschlossen ist.

– Die Seite „*PIN/PUK*“:
  Hier können Sie die Default-Einstellungen von PIN und PUK für die Karten vorgeben. Für die PUK stehen die Einstellungen „*kompatibel*“, „*sicher*“ oder „*keine*“ zur Verfügung. Wenn Sie maximale Sicherheit wünschen, wählen Sie hier die Einstellung „*keine*“, allerdings muss dann eine Chipkarte, deren PIN vom User vergessen oder dreimal falsch eingegeben wurde, weggeworfen werden. Wenn Sie damit rechnen müssen, dass dieser Fall gelegentlich auftritt und Sie diese Kosten vermeiden wollen, ist „*sicher*“ eine gute Wahl, hier können Sie mit einem geeigneten Programm (z.B. dem GUI-Admintool oder auch dem Kommandozeilen- Admintool) die Karte wieder „entsperren“, die PUK wir dabei beim Anlegen des Benutzers als zufällige Folge von 11 alphanumerischen Zeichen gewählt. Für maximalen Komfort auf Seiten der User wählen Sie die Einstellung „*kompatibel*“, hiermit wird eine 6-stellige numerische PUK gewählt, die der User mittels seines Kartenterminals selbst eingeben kann, um die Karte wieder zu entsperren. Dies ist die Default-Einstellung. Beachten Sie, dass auch nach dreimaliger falscher Eingabe der PUK die Karte unwiderruflich unbrauchbar wird.

  Für die PIN können sie zwischen numerischer PIN (diese ist immer 6-stellig und kann am SecOVID Reader eingegeben werden) und alphanumerischer PIN wählbarer Länge (diese zu verwenden macht nur Sinn, wenn sie die Einmalpasswörter stets mittel spezieller Zusatzsoftware - wie dem Programm `getpwd`, vgl. Kapitel 3.3 - und einem an einen Computer angeschlossenen Chipkartenleser ohne Tastatur berechnen) wählen.

– Die Seite „*Passwort Generator*“:

  * Unter „*Chipkartentyp*“ können Sie auswählen, ob Sie standardmäßig Chipkarten personalisieren wollen, die 6- oder 8-stellige Einmalpasswörter erzeugen. Beachten Sie jedoch, dass sie spezielle Kartenleser benötigen, um 6-stellige Einmalpasswörter erzeugen zu können. Bedenken sie insbesondere, dass 6-stellige Passwörter ein höheres Sicherheitsrisiko darstellen und deshalb insbesondere nicht in Verbindung mit höheren Suchtiefen verwendet werden sollten.
  * Aktivieren Sie die Checkbox „*Standard Generator benutzen*“, wenn Sie wünschen, defaultmäßig einen von 0 verschiedenen Vorgabewert für den anzulegenden bzw. zu löschenden Passwort-Generator zu erhalten. (Sie können bis zu zehn verschiedene Passwortgeneratoren auf einer Chipkarte anlegen.)
  * Unter „*Passwort Generator*“ können Sie auswählen, welches dieser Vorgabewert sein soll. Dieser Punkt ist nur dann aktiv, wenn zuvor die Checkbox „*Standard Generator benutzen*“ aktiviert wurde.

  - ∗ Unter „*Standard-SoftToken*“ können Sie wählen ob SoftToken Benutzer mit einem Standard-SoftToken oder mit einem SoftToken in Verbindung mit einer kombinierten Server-PIN ausgestattet werden sollen.
  - ∗ Unter „*Generator-typ ändern*“ können Sie den Menüpunkt „Datenbank“ > „Generator-Typ ändern“ aktivieren bzw. deaktivieren.

  Beachten Sie, dass alle Einstellungen, die Sie auf dieser Seite vornehmen, nur die Vorgabewerte beeinflussen. Beim Erzeugen eines Generators Karte haben Sie stets noch Gelegenheit, diese Vorgabewerte zu ändern.
- Die Seite „*Reporting*“:
  Wenn Sie auf dem SecOVID-Server einen Webserver und die entsprechende Software von der CD installieren, können Sie die Datei(en) `ovid.proto<-Datum>` automatisch auswerten lassen und die Ergebnisse dieser Auswertung mit dem Menüpunkt „Report erstellen“ anzeigen lassen (s.u.). Auf dieser Seite können Sie konfigurieren, wie dieser Webserver erreichbar ist.
  - ∗ Unter „*ServerIP*“ und „*Port*“ tragen Sie die IP-Adresse des SecOVID-Servers und den Port ein, unter dem der Webserver erreichbar ist.
  - ∗ Aktivieren Sie „*HTTPS*“, falls Ihr Server auf dem angegebenen Port das HTTPS-Protokoll benutzt.
  - ∗ Wählen Sie bei „*Server-Typ*“ aus, ob Sie einen „*Apache*“-ähnlichen Webserver (mit Unterstützung für Perl) oder einen „*IIS*“-ähnlichen Webserver (mit Unterstützung für ASP) benutzten.
  - ∗ Unter „*Pfad für Export*“ tragen Sie den Pfad ein, unter dem auf dem WebServer die generierten HTML-Seiten gespeichert werden sollen. Dies ist z.B. relevant, wenn Sie später nochmal alte Auswertungen anschauen wollen, ohne Sie erst neu generieren zu müssen.
  - ∗ Unter „*SecOVID-Pfad*“ tragen Sie den Pfad ein, in dem der SecOVID-Server installiert ist, und in dem die Log-Dateien (`ovid.proto`) abgelegt werden.

Beachten Sie, dass eine neue Verbindung mit der Datenbank aufgebaut wird (gegebenenfalls mit einem neuen Administratornamen), nachdem Sie die Änderungen auf diesen Seiten mit „OK“ bestätigt haben. Hierzu müssen Sie wieder die Administrator-Chipkarte einlegen und die PIN am Kartenterminal eingeben.

Bemerkung: Wollen Sie Ihre Datenbasis sichern oder löschen, so verwenden Sie bitte die Tools und Kommandos, die das Betriebssystem zum Kopieren und Löschen von Dateien zur Verfügung stellt (z.B.`cp /etc/SecOvid/ovid_data /etc/SecOvid/ovid_data.old_version`). Beachten Sie dabei, dass `ovid_data.bak` und `ovid_data.new` für den internen Gebrauch reserviert sind und dass Sie SecOVID-Datenbasen nicht durch einfaches Kopieren auf einen Rechner mit anderer Architektur aufspielen können.

- ***Log-Datei anzeigen:***
  Mit diesem Menüpunkt können Sie die Log-Datei `OvidLog.log` anzeigen, in der z.B. bei Import oder Export von Daten oder auch beim automatischen Anlegen von Benutzern eventuelle Fehlermeldungen protokolliert werden. Gegebenenfalls erscheint jeweils ein ausdrücklicher Hinweis auf diese Datei.

- ***Report erstellen:***
  Wie bereits erwähnt können Sie die Logdateien, die SecOVID anlegt, teilweise automatisch auswerten lassen. Dazu installieren Sie einen WebServer auf dem SecOVID-Server sowie die passenden Module aus dem Unterverzeichnis `admin/ReportServer` des Verzeichnisses für Ihre Platform (`linux`, `solaris` oder `win32`) auf der CD. Schließlich konfigurieren Sie wie oben beschrieben das Admintools so, dass es diesen Webserver findet.

  Wenn Sie nach diesen Vorarbeiten nun den Menüpunkt „*Report erstellen*“ auswählen, öffnet sich ein Fenster, in dem Sie auswählen können, für welchen Zeitraum Sie welche Art von Anfragen auswerten wollen und ob Sie nur eine „*Übersicht*“ über die Anzahl solcher Anfragen oder aber eine „*Liste der Anfragen*“ oder „*Beides*“ sehen wollen.

  Nachdem Sie in diesem Fenster auf „*Erstellen*“ geklickt haben, öffnet sich Ihr Webbrowser und zeigt die gewünschten Informationen an.

- ***Beenden:*** Beenden des Programms.

Datenbank:

- ***Ansicht aktualisieren:***
  Der Inhalt der unter „*Konfiguration*“ eingetragenen Datenbasis wird aus der betreffenden SecOVID-Datenbank ausgelesen und angezeigt.

- ***Datenbank importieren:***
  Fügt Benutzer aus einer verschlüsselten Datei im SecOVID-eigenen Binärformat oder aus einer ASCII-Datei im SecOVID-Importformat zur aktuellen SecOVID-Datenbank hinzu. Nach Aktivieren dieses Menüpunktes erscheint ein Fenster, in welchem Sie die zu importierende Datei auszuwählen haben.
  Falls die zu importierende Datei durch einen Export („Datenbank exportieren“) unter Verwendung einer Administrator-Chipkarte entstanden ist, so muss nun zum Importieren *die gleiche Administrator-Chipkarte*, eingelegt und die Karten-PIN am Kartenterminal eingegeben werden. (Die zu importierende Datei liegt nämlich in diesem Fall verschlüsselt vor und kann nur mit der gleichen Administrator-Chipkarte wieder entschlüsselt werden.)
  Eventuelle Fehlermeldungen werden in der Datei `OvidLog.log` protokolliert. Diese wird in dem Verzeichnis angelegt, das Sie auf der Datenbank-Konfigurationsseite ausgewählt haben.

- ***Datenbank importieren (LDAP):***
  Diese Funktion automatisiert das Anlegen neuer Benutzer, indem diese aus einer bereits bestehenden LDAP-Datenbank (z.B. aus Microsofts ADS oder von einem OpenLDAP-Server) übernommen werden. Wenn Sie dieses Feature benutzen wollen, müssen Sie zunächst die Datei `ldap.conf` anlegen, in der Sie beschreiben müssen, wie der Benutzername, der Nachname und der Vorname, die in der SecOvid-Datenbank eingetragen werden sollen, aus den LDAP-Attributen des Benutzers in Ihrer LDAP-Datenbank abgeleitet werden sollen.
  Sobald dies geschehen ist, können Sie nach Auswahl des Menüpunktes in dem Fenster, das sich dann öffnet, spezifizieren, welche Benutzer aus welcher LDAP-Datenbank importiert werden. Die genaue Syntax hierfür ist abhängig vom verwendeten LDAP-Server.

  – Syntax der Datei `ldap.conf`:
    Abgesehen von Leerzeilen und Kommentarzeilen, die mit `#` beginnen, muss die Datei `ldap.conf` genau 3 Zeilen der Form

    ```
    <Variable> = <LDAP-Attributname>[:[^][ALL BUT] <Zahl>]
    ```

    enthalten, wobei die von `[` und `]` eingeschlossenen Teile optional sind. Für `<Variable>` sind dabei die 3 Werte `login`, `name` und `given name` zulässig, `<LDAP-Attributname>` bezeichnet den Namen eines Attributs in Ihrer LDAP-Konfiguration. Dabei wird normalerweise das gesamte Attribut übernommen, wobei im Falle des von `login` automatisch alle Zeichen in Kleinbuchstaben umgewandelt werden. Wünschen Sie diese Umwandlung nicht oder wollen Sie sie auch für `name` oder `given name` verwenden, so können Sie dies durch das Symbol `^` anzeigen.
    Falls das Attribute wie z.B. im Falle des `common name (cn)` aus mehreren Worten besteht, so ist es unter Umständen wünschenswert, nicht alle Wörter zu übernehmen. Dazu kann mit `<Zahl>` angegeben werden, das wievielte Wort des Attributwertes benutzt werden soll, bzw. mit `ALL BUT <Zahl>`, das wievielte Wort des Attributwertes weggelassen werden soll. Dabei bezeichnen 1,2,3,... das erste, zweite, dritte,... Wort, während -1,-2,-3,... das letzte, vorletzte, drittletzte,... Wort bezeichnen.
    Zur Illustration wollen wir zwei Beispiele geben:
    Im Falle der Kommunikation mit einem ADS von Microsoft können Sie mit

    ```
    login = sn
    name = name
    given name = givenName
    ```

    einfach die Attribute `sn`, `name` und `givenName` als Benutzername, Name und Vorname übernehmen.
    Im Falle der Kommunikation mit einem OpenLDAP-Server, der für jeden Eintrag z.B. die Attribute `sn` und `cn` (also Nachname und vollständiger Name) zurückliefert, können Sie z.B. folgende Konfigurationsdatei verwenden:

```
login = sn
name = cn:-1
given name = cn:1
```

Damit erhalten Sie z.B. für den User mit `cn = Max Philipp Mustermann` dann `Mustermann` als Name (das letzte Wort des cn) und `Max` als Vorname (das erste Wort). Wenn Sie als Vornamen in diesem Beispiel lieber `Max Philipp` extrahieren wollen, so können Sie dazu die Zeile

```
given name = cn:ALL BUT -1
```

verwenden, mit der alles außer dem letzten Wort (dem Nachnamen) als Vorname eingetragen wird.

– LDAP-Anfrage:
Für die eigentliche LDAP-Anfrage müssen sie nun die IP-Adresse und den Port (normalerweise 389) des LDAP-Servers angeben und je nachdem, wie die Zugriffsrechte auf den LDAP-Server konfiguriert sind, müssen Sie auch noch einen Usernamen und ein Passwort angeben, mit denen Sie sich dem LDAP-Server gegenüber authentifizieren.
Achtung: Diese (normalerweise nicht sicherheitskritische) Passwort wird derzeit im Klartext angezeigt.

Achtung: Verwenden Sie Microsofts ADS als LDAP-Server, dann ist als Username *nicht* wie bei einem normalen Login der Benutzername anzugeben, sondern der sog. *Anzeigename*!

Schließlich können Sie in zwei Text-Fenstern der Eingabe-Maske spezifizieren, welche Benutzer Sie importieren wollen. Hier ist zum einen der Suffix des `distinguished name (DN)` anzugeben, den die zu importierenden Benutzer im LDAP-Server haben sollen und zum zweiten ein Suchfilter. Der Suffix ist abhängig von Ihrer LDAP-Konfiguration und kann z.B. die Form `ou=Verkauf,o=Demo GmbH,c=Germany` oder `dc=demo,dc=de` oder ähnliches haben. Für Microsofts ADS ist die letztere Form gängig, eine Domäne `verkauf.demo.de` würde man dort mit `dc=verkauf,dc=demo,dc=de` spezifizieren. Für den Suchfilter können Sie im wesentlichen beliebige Suchfilter nach RFC 2254 verwenden. Jedoch ist zu beachten, dass hier je nach LDAP-Server Einschränkungen gelten können oder leichte Änderungen der Syntax der Anfrage erforderlich sein können. Im Falle von OpenLDAP können Sie z.B. mit `(objectClass=Person)` angeben, dass sie alle Benutzer importieren wollen, für die das Attribut objectClass den Wert `Person` hat (damit lassen sich z.B. administrative Einträge ausfiltern). Im Falle des IPlanet LDAP-Servers müssen Sie dieselbe Anfrage als `(&(objectClass=Person))` codieren, während im Falle von Microsofts ADS dieses Standardattribut einen abweichenden Namen hat, so dass Sie dort für denselben Zweck `(&(objectCategory=person))` verwenden müssen.
Beachten Sie, dass die in dieser Maske eingetragenen Werte (bis auf das eventuell vorhandene Passwort) in der Datei `ldap.cfg` gespeichert werden.

- ***Datenbank exportieren:***
Exportiert die selektierten Benutzer in eine Datei. Diese Export-Datei kann dabei in dem Fenster, das sich nach Auswahl dieses Menüpunktes öffnet, ausgewählt werden. Wenn Sie lokal mit leerem Administratornamen angemeldet sind, dann wird die Exportdatei in einem Klartext-Format exportiert, das ohne große Schwierigkeiten manuell oder maschinell bearbeitet werden kann. Beachten Sie dabei die Doppelpunkte zur Trennung der Felder eines Datenbank-Eintrags voneinander sowie die erste Zeile, die nur aus einem Doppelpunkt besteht und so anzeigt, dass dies eine Klartext-Datei ist.

  Sind Sie mit einem nicht-leeren Administratornamen angemeldet, so müssen Sie Ihre Administrator-Chipkarte einlegen und die PIN am Chipkartenterminal eingeben. In diesem Falle wird die Export-Datei in verschlüsselter Form auf die Platte geschrieben.

- ***Match to LDAP:***
Mit diesem Menüpunkt können Sie noch anonymen („freien SecOVID-Daten“) Chipkarten- oder Token-Daten Einträge aus LDAP- oder ADS-Servern (Active Directory Service) zuordnen.

  Gehen Sie wie folgt vor:

  1. Selektieren Sie die noch anonymen („freien SecOVID-Daten“) Chipkarten- oder Token-Daten-Einträge in der SecOVID-Datenbank.

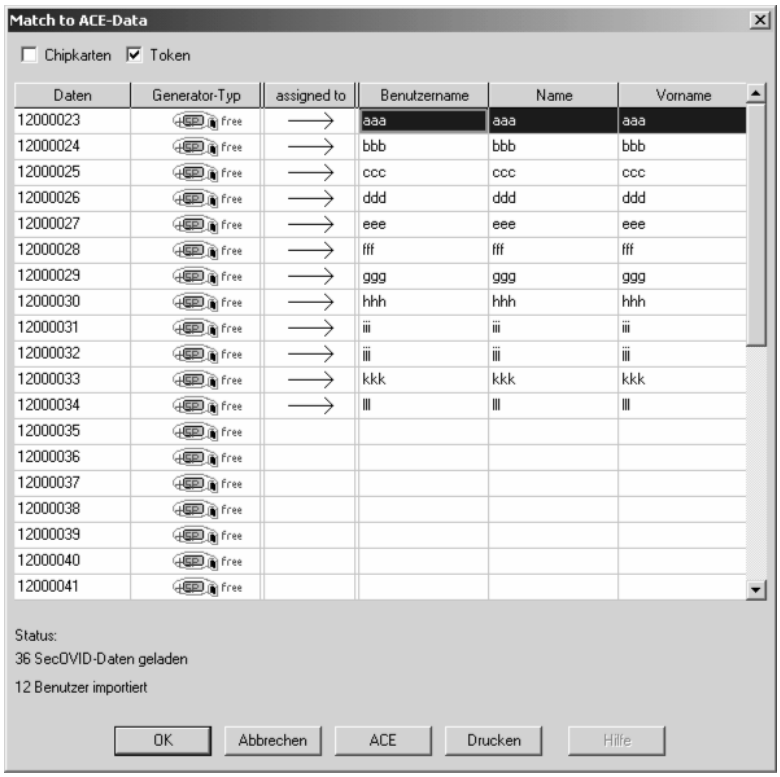
2. Nach Anklicken des Menüpunktes erscheint eine Tabelle, die in den zwei ersten Spalten den Namen und den Generator-Typ der anonymen Daten enthält. Durch Drücken der CheckBoxen „Chipkarte“ bzw. „Token“ wird die SecOVID-Datenbank nach weiteren „freien SecOVID-Daten“ durchsucht, und die ersten beiden Spalten der Tabelle werden ergänzt.
3. Durch Drücken des „LDAP-Buttons“ erscheint eine Eingabe-Maske, mit der Sie spezifizieren können, welchem Benutzer ein Datensatz zugewiesen werden soll. Informationen zu der Syntax der in der Eingabe-Maske erwarteten Daten finden Sie im Menüpunkt „Datenbank importieren (LDAP)“.
4. Den aus dem LDAP-Server ausgelesenen Benutzern wird jeweils automatisch ein freier SecOVID-Datensatz zugeordnet (Vorschlag).
5. Den automatisch generierten Vorschlag können Sie durch folgende Tastenkombinationen verändern.
   - Löschen
     Durch Drücken der „Entf-Taste“ werden die selektierten Einträge aus der Zuordungstabelle gelöscht.
   - Ausschneiden und Einfügen
     Markieren Sie die zu verschiebenden Datensätze, drücken Sie „Strg + x“, klicken Sie auf die gewünschte Einfügestelle in der Tabelle und drücken Sie nun „Strg + v“.
   - Vertauschen der Einträge
     Markieren Sie die zu vertauschenden Datensätze, drücken Sie „Strg + c“, klicken Sie auf die Stelle in der Tabelle dessen Datensätze mit den markiereten Datensätze vertauscht werden sollen und drücken Sie nun „Strg + v“.
6. Durch drücken des „Ok-Buttons“ werden die Zuordnungen in der SecOVID-Datenbank eingetragen. „Abbrechen“ schließt den Dialog, ohne dass eine Zuordnung übernommen wird. Durch drücken des „Drucken-Buttons“ werden alle Zuordnungen in folgender Form ausgedruckt.

```
Login       Lastname    Firstname  OTP-Generator  OTP-Type
aaa         aaa         aaa        000000001      Token + Server-PIN
bbb         bbb         bbb        000000002      Token + Server-PIN
ccc         ccc         ccc        000000003      Token + Server-PIN
ddd         ddd         ddd        000000004      Token + Server-PIN
```

- ***Match to ACE:***
  Mit diesem Menüpunkt können Sie noch anonymen („freien SecOVID-Daten“) Chipkarten- oder Token-Daten Einträge aus der Datenbank des ACE/ Servers zuordnen. Sie benötigen diesen Menüpunkt für eine komfortable Migration von Benutzern des RSA ACE/ Server-Systems (secuID Tokens) auf das SecOVID-System. Hierzu haben Sie jedem zu migrierenden Benutzer ein freies SecOVID-Tokendaten (oder Chipkartendaten) zuzuweisen. (Der Benutzer muss sein SecOVID Token erst benutzen, wenn das RSA securID Token nicht mehr benutzbar ist.)

  Vorgehensweise zur Migration von Benutzern:

  1. Selektieren Sie die noch anonymen („freien SecOVID-Daten“) Chipkarten- oder Token-Daten Einträge aus der Datenbank.
  2. Nach Anklicken des Menüpunktes erscheint eine Tabelle, die in den zwei ersten Spalten den Namen und den Generator-Typ, der anonymen Daten enthält. Durch Drücken der CheckBoxen „Chipkarte“ bzw. „Token“ wird die SecOVID-Datenbank nach weiteren „freien SecOVID-Daten“ durchsucht, und die ersten zwei Spalten der Tabelle werden ergänzt.
  3. Durch Klicken des „ACE-Buttons“ wählen Sie die RSA/ACE-Daten aus (welche Sie vorher aus der ACE/ Server-Datenbank exportiert und mit dem Tool `ace2secovid` konvertiert hatten), die den SecOVID-Daten zugewiesen werden sollen.
  4. Die ausgewählten Datensätze werden automatisch einem SecOVID-Datensatz zugeordnet (Vorschlag).



5. Den automatisch generierten Vorschlag können Sie durch folgende Tastenkombinationen verändern.
   - Löschen
     Durch Drücken der „Entf-Taste“ werden die selektierten Einträge aus der Zuordungstabelle gelöscht.
   - Ausschneiden und Einfügen
     Markieren Sie die zu verschiebenden Datensätze, drücken Sie „Strg + x“, klicken Sie auf die gewünschte Einfügestelle in der Tabelle und drücken Sie nun „Strg + v“.
   - Vertauschen der Einträge
     Markieren Sie die zu vertauschenden Datensätze, drücken Sie „Strg + c“, klicken Sie auf die Stelle in der Tabelle dessen Datensätze mit den markiereten Datensätze vertauscht werden sollen und drücken Sie nun „Strg + v“.
6. Durch Drücken des „Ok-Buttons“ werden die Zuordnungen in der SecOVID-Datenbank eingetragen. „Abbrechen“ schließt den Dialog, ohne dass eine Zuordnung übernommen wird. Durch Drücken des „Drucken-Buttons“ werden alle Zuordnungen in folgender Form ausgedruckt.

```
Login       Lastname    Firstname  OTP-Generator  OTP-Type
aaa         aaa         aaa        000000001      Token + Server-PIN
bbb         bbb         bbb        000000002      Token + Server-PIN
ccc         ccc         ccc        000000003      Token + Server-PIN
ddd         ddd         ddd        000000004      Token + Server-PIN
```

- ***Personalisierung vorbereiten:***
  Exportiert ebenso wie der Menüpunkt „*Datenbank exportieren*“ die selektierten Benutzer in eine Datei. Im Gegensatz zum Menüpunkt „*Datenbank exportieren*“ werden bei Auswahl dieser Funktion jedoch neue Initialisierungsdaten für jeden ausgewählten Benutzer erzeugt. Damit kann man an einer geeigneten Personalisierungsmaschine sofort neue Karten für die selektierten Benutzer erzeugen. Ist allerdings irrtümlich ein Benutzer selektiert, der bereits eine Karte besitzt, so wird seine alte Karte ungültig, falls Sie in dem Rückfrage-Fenster bestätigen, dass Sie für diesen Benutzer tatsächlich eine neue Karte erzeugen wollen.

Wie beim Export aus der Datenbank wird die erzeugte Datei entweder Klartext-Format haben oder verschlüsselt sein. Gegebenenfalls müssen Sie insbesondere die Administrator-Chipkarte einlegen und die PIN eingeben.

- ***Drucke PS-Formular:***
  Druckt die ausgewählten Benutzer-Daten automatisch in ein PIN-Brief-Formular. Nach Auswahl dieses Menüpunktes erscheint ein Fenster, in dem Sie auswählen können, welche Daten Sie für die ausgewählten Benutzer ausdrucken wollen, und wo diese auf dem Papier platziert werden sollen. Für die Platzierung sind jeweils der gewünschte Abstand vom oberen und vom linken Seitenrand anzugeben. Konfigurieren Sie Ihren Drucker so, dass er eine Postscript-Testseite mit den korrekten Abständen vom Rand ausdruckt. In diesem Fall wird auch das genannte PIN-Brief-Formular korrekt auf dem Papier platziert.

- ***Benutzer Information:***
  Sie erhalten ein Dialogfenster mit den folgenden Informationen zum Benutzer:
  - den Benutzernamen *(editierbar)*
  - Name und Vorname *(editierbar)*
  - weitere Generatoren
  - die PIN,
    - die auf die Karte des Benutzers geschrieben wurde bzw. beim Anlegen eines Generators auf die Karte geschrieben wird *(editierbar)*
    - die bei der Verwendung eines Tokens mit PIN-Schutz eingegeben werden muss
    - die als Server-PIN verwendet wird
  - die PUK, die auf die Karte geschrieben wurde bzw. beim Anlegen eines Generators auf die Karte geschrieben wird
  - die Seriennummer des Tokens bzw. der Chipkarte
  - der Generator, der vom Benutzer zum Erzeugen eines Einmalpasswort verwendeten wird.
  - die Gruppe(n) in der der Benutzer Mitglied ist.
  - wann die Karte/Token erstellt wurde
  - wann der Benutzer sich beim SecOVID-Server zuletzt angemeldet hatte
  - das letzte Einmalpasswort des Benutzers
  - die SecOVID-Server-Systemzeit, zu welcher der ggf. für eine gewisse Zeitspanne gesperrte Benutzer sich wieder mit korrektem Einmalpasswort authentifizieren kann.
  - unter „Status“: ob der Benutzer gesperrt wurde oder nicht. (Im letzten Fall gibt z.B. „2 Fehler“ die Anzahl der Fehlversuche (hier: 2) seit dem letzten erfolgreichen Einloggen an.)

- ***Neuer Benutzer:***
  Sie erhalten ein Menü, in welchem Sie Benutzername, Name und Vorname des neu anzulegenden Benutzers eintragen. Durch das Markieren einer oder mehrerer Gruppen in der Auswahlbox, wird dem Benutzer die selektierte(n) Gruppe(n) zugewiesen. Die Auswahl der Gruppen beschränkt sich die, für die der Administrator zuständig ist. Wenn das Kontrollfeld „*PIN autom.*“ gesetzt ist, wird für den Benutzer automatisch eine PIN erzeugt und im betreffenden Textfeld angezeigt. Ist das Häkchen „*PIN autom.*“ nicht gesetzt, können Sie jetzt selbst eine PIN definieren, die später bei „*Karte erzeugen*“ verwendet wird. Sie können dieses Feld auch leer lassen, um erst im letzten Moment eine PIN zu definieren, wenn nämlich ein Passwortgenerator auf eine leere oder bereits benutzte Chipkarte geschrieben werden soll. (Diese PIN-Eingabe im letzten möglichen Moment könnte dann auch durch den Benutzer erfolgen, sofern dieser vor Ihnen steht). Für die eingegebene PIN findet keine Längenprüfung statt. Beachten Sie, dass der SecOVID Reader (Plus) im offline-Modus nur sechsstellige PINs (im Falle der älteren TCOS 1.2-Karten alternativ vierstellige PINs) verarbeiten kann. Beachten Sie, dass Sie mittels einer kleinen Zusatzbibliothek lokale Regeln für die Gestaltung des Benutzernamens überwachen können, siehe hierzu Abschnitt 3.1.2.7. Beachten

Sie, dass in keiner der eingegebenen Zeichenfolgen, also weder im Benutzernamen, noch im Namen oder Vornamen oder in der PIN, Doppelpunkte vorkommen dürfen. Darüber hinaus sind im Benutzernamen auch Leerzeichen sowie die beiden Zeichen `@` und `/` verboten.

- ***Benutzer anlegen (autom.):***
  Dieser Menüpunkt dient zum automatischen Generieren von Benutzern. Von Interesse ist dieser Menüpunkt, wenn Sie viele Karten anonym personalisieren möchten und erst später - etwa an einer separaten Ausgabestelle - die Zuordnung von Karte zu Benutzern vornehmen. Sie können z.B. die Benutzer „*Test0000*“ bis „*Test1500*“ erzeugen, indem Sie als Benutzer-ID „*Test*“ angeben und für die Zahlen a und b „*0*“ und „*1500*“ eingeben. Durch das Markieren einer oder mehrerer Gruppen in der Auswahlbox, wird den Benutzern die selektierte(n) Gruppe(n) zugewiesen. Die Auswahl der Gruppen beschränkt sich die, für die der Administrator zuständig ist. Wenn Sie „*autom. PIN*“ selektieren, wird für jeden Benutzer automatisch eine zufällige PIN erzeugt. Statt dessen können Sie jetzt für jeden Benutzer die gleiche PIN definieren, indem Sie das Häkchen in „*autom. PIN*“ entfernen. Ferner können Sie durch Entfernen des Häkchens in „*autom. PIN*“ die Karten-PINs der Benutzer erst im letzten möglichen Moment definieren; lassen Sie hierzu das PIN-Feld leer.
  Beachten Sie bei Verwendung dieses Menüpunkts insbesondere, dass es nicht möglich ist, mehr Benutzer zu erzeugen, als Ihre Lizenz gestattet.

- ***Benutzer sperren:***
  Hiermit sperren Sie die selektierten Benutzer, und der Status des Benutzers wechselt auf „-1“. Das bedeutet, dass die betreffenden User bis zu ihrer Entsperrung keinen Einlass ins System erhalten.

- ***Benutzer entsperren:***
  Hiermit entsperren Sie die selektierten Benutzer, und der Status der Benutzer wechselt auf „0“. Das bedeutet, dass die betreffenden User wieder mit einem korrekten Einmalpasswort Einlass ins System erhalten.

- ***Benutzer löschen:***
  Löschen von Benutzern. Voraussetzung zum Löschen: Die entsprechenden Benutzer müssen gesperrt sein. Beachten Sie, dass durch den Löschvorgang die Karte des Benutzers völlig unbrauchbar wird, falls sie nicht zuvor gelöscht wurde. Nach dem Löschen des Benutzers aus der Datenbank, kann die Karte nicht mehr gelöscht werden, da die dazu notwendige PUK nirgendwo mehr gespeichert ist. (Im Falle von TCOS-1.2- oder CardOS-Karten, ist ein Löschen noch möglich, falls der Benutzer seine PIN mitteilt - bei diesen Karten ist die PIN ausreichend, um die Karte löschen zu können. In diesem Falle können Sie einen neuen Benutzer mit derselben PIN anlegen, diesen auswählen, und dann den Menüpunkt Applikation löschen auswählen.)

- ***Benutzer suchen:***
  Suchen nach Benutzern in der Datentabelle. Dabei gibt es zwei verschiedene Suchmodi, die „*Schnellsuche*“ und die „*Komplexe Suche*“. Bei Verwendung der „*Komplexen Suche*“ sind die möglichen Suchkriterien umfangreicher, allerdings kann diese bei großen Datenbanken lange dauern. Nach Auswahl des Menüpunktes erscheint ein Fenster, in welchem Sie angeben können,

  – nach welchen Muster in der Datentabelle gesucht werden soll.
    Sie können nach mehreren Mustern/Benutzern gleichzeitig suchen, verschiedene Muster werden dabei durch Leerzeichen getrennt, das Muster darf aus beliebigen alphanumerischen Zeichen bestehen. Eine Sonderbehandlung erfahren dabei die Zeichen ’*’ und ’?’. Das Zeichen ’*’ ist ein Platzhalter für eine beliebige Zeichenkette mit beliebiger Anzahl von Buchstaben (Anzahl 0 ist auch möglich). Das Zeichen ’?’ ist ein Platzhalter für ein einzelnes beliebiges Zeichen. Beide Zeichen sind allerdings nur in der „*Komplexen Suche*“ erlaubt, in der „*Schnellsuche*“ löst ihr Verwendung eine Fehlermeldung aus.
    Beispiele:
    ‘???’ findet alle Zeichenketten, die aus drei Zeichen bestehen.
    ‘u*’ findet alle Zeichenketten, die mit ‘u’ beginnen.

  - ob Sie nach Benutzernamen (Benutzer-ID), Namen oder Vornamen suchen wollen. Klicken Sie zur Aktivierung oder Deaktivierung des Suchbereichs auf das entsprechende Kontrollfeld. Mehrfachnennungen sind möglich und werden durch „ODER“ verknüpft.
  - ob Sie bei „*Komplexer Suche*“ nach Gruppen suchen wollen.
  - ob die Suche nach Benutzernamen, Namen oder Vornamen die Groß- und Kleinschreibung berücksichtigen soll oder nicht (nur bei „*Komplexer Suche*“). Damit können Sie also z.B. einstellen, dass ‘U*’ alle Benutzer finden soll, deren Benutzername mit ‘u’ oder ‘U’ beginnt oder deren Vorname mit ‘u’ oder ‘U’ beginnt.

Die Benutzereinträge, die als Ergebnis der Suche gefunden wurden, sind anschließend zusätzlich zu den vorher bereits selektierten Benutzern selektiert.

- ***Benutzer resynchronisieren:***
  Resynchronisieren von Benutzern. Voraussetzung zum Resynchronisieren: Die entsprechenden Benutzer müssen gesperrt sein. Hiermit wird individuell für einen Benutzer die übliche Suchtiefe um einen Faktor 10 erhöht. Der Benutzer muss dann allerdings zweimal hintereinander versuchen, sich anzumelden, wobei aufeinanderfolgende Einmalpasswörter angegeben müssen. Damit wird gewährleistet, das die Resynchronisation tatsächlich korrekt abläuft, es könnte innerhalb eines sehr großen Suchintervalls ja ein Passwort womöglich zweimal vorkommen. Auch wird verhindert, dass ein Unbefugter hier bessere Chancen hat, durch Raten eines Passwortes in das System einzudringen. In der Statusspalte erscheint für einen Benutzer, den Sie gerade zur Resynchronisation ausgewählt haben, der Status „*Resync(1)*“. Sobald der Benutzer das erste der beiden Passwörter eingegeben hat, ändert sich der Status auf „*Resync(2)*“. Nach Eingabe des korrekten zweiten Passwortes ändert sich der Status wieder auf „*OK*“. Ist das zweite Passwort nicht das unmittelbar auf das erste folgende, so ändert sich der Status auf „*gesperrt*“.
  Beachten Sie, dass Aliasnamen (siehe Beschreibung des Menüpunkts *Aliasnamen hinzufügen/löschen*) in besonderer Weise behandelt werden: Wenn Sie für einen Benutzer `testuser` z.B. eine Chipkarte und mit dem Benutzernamen `testuser-token` ein Token personalisiert haben, dann können Sie einerseits wie gewohnt mittels `testuser` die Chipkarte und über den Namen `testuser-token` das Token resynchronisieren. Im allgemeinen, insbesondere wenn die Benutzernamen, unter denen weitere Token eines Benutzers in der Datenbank eingetragen sind, gar nicht bekannt sind, ist dies jedoch nicht sehr praktikabel. Daher ist es in diesem Fall auch möglich, nachdem der Benutzer `testuser` auf resynchronisieren gestellt wurde, sonstige Einmalpasswortgeneratoren dieses Benutzers, in unserem Beispiel also das Token, zu resynchronisieren. Dies funktioniert allerdings nur, wenn sie dann auch wirklich das zu resynchronisierende Token benutzen. Wenn Sie mit einem anderen Einmalpasswortgenerator, z.B. der Chipkarte, ein korrektes Einmalpasswort erzeugen und zur Anmeldung benutzen, geht der Resynchronisationsstatus des Benutzers verloren, um das Token zu resynchronisieren, müssten sie also erneut den Menüpunkt *Benutzer resynchronisieren* für diesen Benutzer auswählen.

- ***Generator-Typ ändern:***
  Ermöglicht das Ändern des Generator-Typs. Diese Feature wird benötigt, wenn Sie einen SecOVID-Datensatz importieren wollen, der unter der SecOVID Version 3.0 erstellt wurde bzw. wenn Sie ein Upgrade von SecOVID 3.0 auf SecOVID 3.1 machen. Sie benötigen diesen Menüpunkt insbesondere, um zu definieren, ob der SecOVID-Server für einen Benutzer zusätzlich zu einem Einmalpasswort auch dessen statisches Passwort (Server-PIN) prüfen soll.

  Mögliche Generatoren-Typen:

  - undefiniert
  - Chipkarte
    Als Passwort hat der Benutzer einzugeben: `<OTP>`
  - SecOVID Token + PIN-Schutz (nicht mehr unterstützt)
    Als Passwort hat der Benutzer einzugeben: `<OTP>`

- SecOVID Token + Server-PIN
  Als Passwort hat der Benutzer einzugeben: `<Server-PIN><OTP>`
- SecOVID Token + PIN-Schutz + Server-PIN (nicht mehr unterstützt)
  Als Passwort hat der Benutzer einzugeben: `<Server-PIN><OTP>`
- SecOVID Token
  Als Passwort hat der Benutzer einzugeben: `<OTP>`
- SecOVID SoftToken
  Als Passwort hat der Benutzer einzugeben: `<OTP>`
- SecOVID SoftToken + Server-PIN
  Als Passwort hat der Benutzer einzugeben: `<Server-PIN><OTP>`

Die SecOVID Token Varianten mit PIN Schutz sind nicht mehr verfügbar und daher nur der Vollständigkeit halber aufgeführt.

Zusätzlich können Sie einen SecOVID-Datensatz als „free“ markieren, wenn Sie diesen Passwortgenerator einem Benutzer zuweisen wollen, dessen Personalien (Benutzername, Nachname, Vorname) durch die Funktion „Match to LDAP“ oder „Match to ACE“ aus einer fremden Datenquelle importiert werden.

Bemerkung:
Dieser Menüpunkt muss über die Menüpunkte „Programm“ > „Konfiguration“ > „ Paßwortgenerator “ frei geschaltet werden.

- ***Server-PIN ändern:***
  Bei Verwendung eines Passwortgenerators mit Server-PIN (z.B. SecOVID Token mit Server-PIN) kann dieser Menüpunkt genutzt werden, um die aktuelle Server-PIN für einen Benutzer zu ändern. In dem erscheinenden Dialogfeld muss die Server-PIN zwei Mal eingegeben werden.

- ***Administratoren-Datenbank:***
  Hiermit können Sie zwischen der Verwaltung von Benutzern und Administratoren hin- und herschalten. Die Verwaltung von Administratoren benötigen Sie insbesondere, um einen neuen Administrator und eine neue Administratorkarte zu erzeugen.

- ***Ausgewählte Benutzer/Alle Benutzer anzeigen:***
  Hiermit können Sie zwischen den beiden möglichen Ansichten der Datenbank hin- und herschalten, so dass entweder nur die gerade ausgewählten oder alle Benutzer angezeigt werden. Beachten Sie dass die Suchfunktion automatisch in die Ansicht umschaltet, in der nur die ausgewählten Benutzer angezeigt werden.

Chipkarten:

- ***Karte erzeugen***
  Anlegen eines Einmalpasswortgenerators auf der Karte.
  Selektieren Sie zunächst die Benutzer, auf deren Karten Sie Einmalpasswortgeneratoren schreiben wollen. Klicken Sie dann den genannten Menüpunkt an. Die selektierten Benutzer werden nun der Reihe nach abgearbeitet. In einem Fenster sehen Sie jeweils, für welchen Benutzer nun ein Einmalpasswortgenerator auf eine Karte geschrieben werden soll. Sie können jeweils durch Anklicken des „OK“-Buttons den Schreibvorgang für den angezeigten Benutzer akzeptieren, durch Anklicken von „Überspringen“ auf das Anlegen eines Einmalpasswortgenerators für den angezeigten Benutzer verzichten und zum nächsten selektierten Benutzer übergehen und durch Anklicken von „Abbrechen“ den gesamten Vorgang des Anlegens von Einmalpasswortgeneratoren abbrechen.
  Vor dem Akzeptieren des Anlegens eines Einmalpasswortgenerators auf einer Karte wählen Sie unter „Passwortgenerator“ jeweils die Nummer des Passwortgenerators aus, der auf der Karte angelegt werden

soll. Auf jeder Karte können bis zu zehn Passwortgeneratoren gleichzeitig existieren. Diese haben die Nummern 0, 1, 2, ..., 9. Passwortgeneratoren können einzeln und unabhängig voneinander angelegt und gelöscht werden. Durch dieses Feature können z.B. Doppelrollen realisiert werden, und die Daten verschiedener Passwortgeneratoren auf der Karte können in unterschiedlichen SecOVID-Servern registriert sein. Außerdem können Sie hier auch einstellen, ob eine Karte zur Erzeugung 6-stelliger oder 8-stelliger Passwörter erzeugt werden soll. Beachten Sie nochmals, dass zur ERzeugung 6-stelliger Passwörter spezielle Chipkartenterminals benötigt werden.

Beispiel:
Der Systemverwalter Klaus Schmidt könnte eine Karte besitzen, auf der die Passwortgeneratoren 0, 1 und 2 existieren, während Passwortgeneratoren 3 bis 9 nicht auf der Karte vorhanden sind. Passwortgenerator 0 produziert dabei Einmalpasswörter, die vom Internet-Service-Provider von Klaus Schmidt sowie von dessen Hausbank beim Online-Banking akzeptiert werden (Authentifikation durch das Trust Center der Deutschen Telekom). Die Einmalpasswörter von Passwortgenerator 1 berechtigen zum Remotezugriff auf das Firmennetz unter dem Benutzernamen schmidt, während die Passwörter von Generator 2 zum entsprechenden Zugang für den Benutzer root berechtigen. Die Person Klaus Schmidt besitzt also eine einzige Karte, die Passwörter für verschiedene Zwecke (Einsatzorte) und verschiedene Accounts produziert.

Warnung:

1. Falls Sie den Einmalpasswortgenerator des Benutzers schmidt auf einer Karte anlegen möchten, die bereits einen Passwortgenerator enthält, so muss der Benutzer schmidt in der Datentabelle die PIN dieser Karte besitzen bzw. falls in der Datentabelle für den Benutzer schmidt keine PIN registriert ist, muss jetzt exakt die PIN der Karte angegeben werden. Andernfalls wird der Schreibvorgang scheitern (siehe unten).
2. Das Trust Center der Deutschen Telekom benutzt für seine Karten den Passwortgenerator 0. Wenn Sie Ihren Benutzern also die Möglichkeit offenhalten wollen, sich einen Passwortgenerator dort einrichten zu lassen, so verwenden Sie bitte ausschließlich die Generatoren 1 bis 9.

Nach der Auswahl des Passwortgenerators werden Sie zum Einlegen einer Chipkarte in das Chipkartenterminal (Terminal der Firma KOBIL) aufgefordert. Falls Sie eine Chipkarte in das Chipkartenterminal einlegen, für die noch keine PIN definiert ist (was z.B. bei einer noch unbenutzten Chipkarte der Fall ist), so wird nun die PIN-Nummer auf der Karte als PIN angelegt, die für den Benutzer vorab definiert wurde. (Ein Rechtsklick auf den Benutzereintrag zeigt die definiert PIN an). Wurde vorab keine PIN definiert, so müssen Sie oder der Benutzer jetzt die Karten-PIN eingeben. Wenn Sie eine Chipkarte in das Chipkartenterminal einlegen, auf der bereits ein PIN-File vorhanden ist, so stellen Sie sicher, dass die für den Benutzer vorab registrierte oder jetzt einzugebende PIN mit der PIN auf der Karte übereinstimmt; ansonsten wird der Versuch, einen Einmalpasswortgenerator auf die Karte zu schreiben, scheitern.
Der entsprechende SecOVID-Passwortgenerator wird nun angelegt. Die erforderlichen Zufallswerte (Triple-DES-Schlüssel und initiales Passwort/ Ciphertext) werden dabei durch den Zufallszahlengenerator der Chipkarte erzeugt und auf Chipkarte sowie in der Datenbank abgespeichert. Je nachdem, ob betreffende Nummern schon auf der Karte vorliegen oder nicht, werden die in der Datenbank für den Benutzer registrierten Werte für PIN und PUK auf die Karte geschrieben. Die PUK (Personal Unblocking Key) kann später sozusagen als Super-PIN dazu benutzt werden, um eine neue PIN auf der Karte einzutragen, wenn der Benutzer seine PIN vergessen haben sollte (s.u.).

Warnung:
Falls Sie eine Karte einlegen, die bereits einen Einmalpasswortgenerator, also bereits eine PUK enthält, so bleibt die PUK auf dieser Karte erhalten. Sie ist nicht von der Karte auslesbar, kann daher nicht in die Datenbank eingetragen werden. Die Folge: Sie können die Karten-PIN unter Verwendung dieses Benutzer-Eintrags nicht mehr zurücksetzen (wenn die PIN z.B. wiederholt falsch eingegeben wurde und deswegen gesperrt ist). Das Löschen eines Einmalpasswortgenerators oder der gesamten SecOVID-Applikation (alle Einmalpasswortgeneratoren) auf der Karte ist dennoch bei Kenntnis der Karten-PIN möglich. Auf TCOS-

2.0-Karten bleibt in diesem Falle allerdings die PUK zurück, die nur mit Kenntnis der PUK entfernt werden kann.

Hinweis:
Mit diesem Menüpunkt legen Sie insbesondere auch die Chipkarten für SecOVID-Administratoren an. Beachten Sie in diesem Fall, dass für die Administratoren stets automatisch der Einmalpasswort-Generator Nr. 9 verwendet wird.

- ***Key Generator löschen***
  Löschen eines einzelnen Einmalpasswortgenerators auf der SecOVID-Chipkarte. Alle übrigen Daten wie PIN, PUK, sonstige Passwortgeneratoren oder sonstige nicht-SecOVID-spezifische Daten verbleiben auf der Karte.
  Selektieren Sie zunächst die Benutzer, auf deren Karten Sie Einmalpasswortgeneratoren löschen wollen. Klicken Sie dann den genannten Menüpunkt an. Die selektierten Benutzer werden nun der Reihe nach abgearbeitet. In einem Fenster sehen Sie jeweils, für welchen Benutzer nun ein Einmalpasswortgenerator auf einer Karte gelöscht werden soll. Sie können jeweils durch Anklicken des „OK“-Buttons den Löschvorgang für den angezeigten Benutzer akzeptieren, durch Anklicken von „Überspringen“ auf das Löschen eines Einmalpasswortgenerators für den angezeigten Benutzer verzichten und zum nächsten selektierten Benutzer übergehen und durch Anklicken von „Abbrechen“ den gesamten Vorgang des Löschens von Einmalpasswortgeneratoren abbrechen.
  Vor dem Akzeptieren des Löschen eines Einmalpasswortgenerators auf einer Karte wählen Sie unter „Passwortgenerator“ jeweils die Nummer des Passwortgenerators aus, der auf der Karte gelöscht werden soll. Sie werden schließlich zum Einlegen der Chipkarte des betreffenden Benutzers in das Chipkartenterminal (Terminal der Firma KOBIL) aufgefordert. Zum Löschen muss die PIN der Karte mit dem in der Datenbank für den Benutzer registrierten Wert übereinstimmen. Wurde vorab keine PIN definiert, ist also für den Benutzer keine PIN in der Datenbank registriert, so müssen Sie oder der Benutzer jetzt die Karten-PIN eingeben. (Falls die PIN-Werte nicht übereinstimmen, fordern Sie den Benutzer zu einer entsprechenden PIN-Änderung mittels des SecOVID Readers (Plus) auf. Falls die korrekte PUK der Karte in der Datenbank gespeichert ist (das ist immer dann der Fall, wenn die Karte beim Anlegen des betreffenden Passwortgenerators vollkommen leer war), können Sie alternativ zunächst die Karten-PIN zurücksetzen und dann versuchen, den Passwortgenerator zu löschen.)

- ***Applikation löschen***
  Löschen der gesamten SecOVID-Applikation mit allen SecOVID-Einmalpasswortgeneratoren (inkl. PIN und PUK) von der SecOVID-Chipkarte.
  Selektieren Sie zunächst die Benutzer, auf deren Karten Sie alle Einmalpasswortgeneratoren löschen wollen. Klicken Sie dann den genannten Menüpunkt an. Die selektierten Benutzer werden nun der Reihe nach abgearbeitet. In einem Fenster sehen Sie jeweils, für welchen Benutzer nun alle Einmalpasswortgeneratoren auf einer Karte gelöscht werden sollen. Sie können jeweils durch Anklicken des „OK“-Buttons den Löschvorgang für den angezeigten Benutzer akzeptieren, durch Anklicken von „Überspringen“ auf das Löschen aller Einmalpasswortgeneratoren für den angezeigten Benutzer verzichten und zum nächsten selektierten Benutzer übergehen und durch Anklicken von „Abbrechen“ den gesamten Vorgang des Löschens von SecOVID-Applikationen abbrechen. Sie werden schließlich zum Einlegen der zu löschenden Chipkarten in das Chipkartenterminal (Terminal der Firma KOBIL) aufgefordert. Zum Löschen muss die PIN der Karte mit dem in der Datenbank für den Benutzer registrierten Wert übereinstimmen. Wurde vorab keine PIN definiert, ist also für den Benutzer keine PIN in der Datenbank registriert, so müssen Sie oder der Benutzer jetzt die Karten-PIN eingeben. (Falls die Werte nicht übereinstimmen, fordern Sie den Benutzer zu einer entsprechenden PIN-Änderung mittels des SecOVID Readers auf. Falls die korrekte PUK der Karte in der Datenbank gespeichert ist (das ist immer dann der Fall, wenn die Karte beim Anlegen der SecOVID-Applikation vollkommen leer war), können Sie alternativ zunächst die Karten-PIN zurücksetzen und dann versuchen, die SecOVID-Applikation zu löschen.)

- ***PIN zurücksetzen***
  Auf der SecOVID-Chipkarte wird protokolliert, wie oft die falsche Karten-PIN eingegeben wurde. Wenn der entsprechende Zähler (Fehlbedienungszähler) eine gewisse Schranke erreicht, sperrt sich die Karte

automatisch. Eine Entsperrung, verbunden mit einem Neueintragen der PIN, ist nur unter Angabe der PUK (Personal Unblocking Key) möglich. Durch Anklicken des genannten Eintrags wird die betreffende PUK aus der Datenbank ausgelesen und die PIN auf die Nummer zurückgesetzt, die für den betreffenden Benutzer in der Datenbank gespeichert ist. Ist für den Benutzer keine PIN in der Datenbank registriert, so erscheint die Aufforderung, jetzt eine neue PIN für die Karte zu definieren. Kommen Sie (oder der Benutzer) dieser Aufforderung bitte nach.
Selektieren Sie zunächst die Benutzer, auf deren Karten Sie die PIN zurücksetzen möchten. Klicken Sie dann den genannten Menüpunkt an. Die selektierten Benutzer werden nun der Reihe nach abgearbeitet. In einem Fenster sehen Sie jeweils, für welchen Benutzer nun die PIN auf der Karte zurückgesetzt werden soll. Sie können jeweils durch Anklicken des „OK“-Buttons den Vorgang für den angezeigten Benutzer akzeptieren, durch Anklicken von „Überspringen“ auf das Durchführen des Vorgangs für den angezeigten Benutzer verzichten und zum nächsten selektierten Benutzer übergehen und durch Anklicken von „Abbrechen“ den gesamten Vorgang des Zurücksetzens der PIN abbrechen. Sie werden schließlich zum Einlegen der Chipkarte des aktuell angezeigten Benutzers in das KOBIL-Chipkartenterminal aufgefordert. (Anschließend müssen Sie eine neue Karten-PIN definieren, sofern für den Benutzer keine PIN in der Datenbank registriert ist.)

Sonstige Token

- ***Tokendaten importieren***
  Fügt die Tokendatensätze zur aktuellen SecOVID-Datenbank hinzu.
  Wenn Sie mit SecOVID Tokens arbeiten wollen, müssen Sie über diesen Menüpunkt einmalig die auf einem Datenträger erhaltenen geheimen Tokendaten in die SecOVID-Datenbank importieren. Handelt es sich bei den SecOVID Tokens um eine Teststellung, so liegen die Tokendaten im Klartext vor und können ohne Weiteres in die SecOVID-Datenbank eingefügt werden. Haben Sie die SecOVID Tokens gekauft, so erhalten Sie die Tokendaten in der Regel verschlüsselt. Beim Versuch, diese Daten zu importieren, werden Sie aufgefordert, Ihre „KOBIL-Kommunikationschipkarte“ in das angeschlossene KOBIL-Chipkartenterminal einzulegen und die Karten-PIN einzugeben. Eine Entschlüsselung der Tokendaten ist nur mit dieser KOBIL-Kommunikationschipkarte möglich. Bewahren Sie sie an einem sicheren Ort auf und merken Sie sich die Karten-PIN! Sie benötigen diese Karte zum Import der Tokendaten nach jeder Bestellung von SecOVID Tokens.

  Bemerkung:
  Wird nach dem Importieren der Tokendatensätze ein Fragezeichen in der Spalte „Generator-Typ“ angezeigt, so müssen Sie den Generator-Typ ändern. (siehe Generator-Typ ändern)

- ***SoftToken erzeugen***
  Anlegen eines SoftTokens (bei Administratoren z.B. für den Remote-Access bei der Verwendung des Kommandozeilentools, bei normalen Benutzern für die Verwendung auf Handys oder PDAs). Für die ausgwählten Benutzer wird das Softtoken jeweils in einer Datei namens *Benutzer-ID*.kst gespeichert. Diese Datei wird in dem Export-Verzeichnis abgelegt, dass Sie auf der Datenbank-Konfigurationsseite ausgewählt haben, siehe . Beachten Sie, dass jeweils die PIN, die in der Datenbank gespeichert ist, benutzt wird, um das File zu verschlüsseln. Aus Sicherheitsgründen sollten sie unbedingt sicherstellen, dass diese PIN ein ausreichend gutes Passwort ist, z.B. indem Sie vor Erzeugung von SoftToken ändern. Das Passwort sollte 8 Zeichen lang sein und kein Wort sein, dass in einem Wörterbuch zu finden ist.

  Bemerkung:
  Dieser Menüpunkt ist nur in der Administratoren-Ansicht aktiviert.

- ***Notfallpasswort erzeugen***
  Anlegen eines vorübergehend gültigen Notfallpassworts für Benutzer, die z.B. ihr Token vergessen haben.

Für die ausgwählten Benutzer wird jeweils in einem eigenen Dialog ein statisches Notfallpasswort gesetzt, mit dem sich der Benutzer anstelle seines üblichen Einmalpassworts anmelden kann. Dieses Passwort ist nur für den im Dialog eingestellten Zeitraum gültig, nach Ablauf der Zeit gilt automatisch wieder nur noch das Einmalpasswort. Das Einmalpasswort wird ebenfalls wieder gültig, falls sich der Benutzer vor Ablauf der Frist unter Verwendung eines Einmalpassworts authentifiziert.

Achtung: Beachten Sie, dass das statische Notfallpasswort die Server-PIN überschreibt, falls eine solche verwendet wird, d.h. nach Ablauf des Notfallpassworts (oder auch bei einer Anmeldung mit Einmalpasswort und Server-PIN vor Ablauf der Gültigkeitsdauer des Notfallpassworts) wird das Notfallpasswort anstelle der bisherigen Server-PIN erwartet. Diese neue Server-PIN kann natürlich in der üblichen Weise geändert werden.

Bemerkung:
Dieser Menüpunkt ist nur in der Administratoren-Ansicht aktiviert.

- ***Aliasnamen hinzufügen/löschen***
  Ordnet die ausgewählten Generatoren einem weiteren, noch auszuwählenden Datensatz als zusätzliche Generatoren zu. Hiermit kann man z.B. einem Benutzer mehrere Generatoren zuordnen oder einen Generator für mehrere Benutzer verwenden. Wenn Sie diesen Punkt auswählen öffnet sich ein Fenster, in dem Sie eingeben können, für welchen Benutzer die Generatoren der ausgewählten Benutzer ebenfalls gültig sein sollen. Nachdem Sie dies getan haben erscheint ein Fenster, das nochmal zeigt, wessen Generatoren Sie jetzt bearbeiten. Im rechten Fenster sind dabei die Generatoren aufgeführt, die bereits für diesen Benutzer gültig sind, im linken Fenster sind die Generatoren aufgeführt, die sie hinzufügen wollen. Um nun einen Generator wirklich für einen Benutzer freizuschalten, müssen Sie ihn vom linken in das rechte Fenster bewegen und können ihn dort mittels der Buttons „Auf“ und „Ab“ einsortieren.

  Achtung: Beachten Sie, dass hierbei die Passwörter jeden Benutzers, der auf der rechten Seite unterhalb eines anderen Benutzers steht, auch für diesen anderen Benutzer akzeptiert werden - der Besitzer des untersten Generators kann sich also mit allen rechts aufgeführten Namen anmelden, soweit diese im System vorhanden sind.

  Beachten Sie außerdem, dass zur Verifikation eines Passworts die Generatoren in der aufgeführten Reihenfolge von oben nach unten durchprobiert werden. Um eine möglichst schnelle Bearbeitung zu gewährleisten und um Komplikationen zu minimieren, die entstehen können, wenn ein Passwort zufällig für mehrere Generatoren gültig sein sollte, sollten Sie die Generatoren nach der Häufigkeit, mit der sie benutzt werden, sortieren, das meistbenutzte zuoberst.

  In den meisten Fällen wird die Reihenfolge mit Blick auf die Zugriffsrechte egal sein, wenn Sie etwa für einen Benutzer `testuser` mit Chipkarte zusätzlich `testuser-token` und/oder `testuser-handy` anlegen und diese beiden `testuser` zuordnen, dann wird es die Benutzer `testuser-handy` bzw. `testuser-token` im System nicht geben, so dass es gleichgültig ist, ob nun die Passwörter von `testuser-handy` auch für `testuser-token` akzeptiert werden oder umgekehrt.

  Anders sieht die Lage aus, wenn Sie es ermöglichen wollen, dass sich Benutzer mit Ihrem Token mit verschiedenen Benutzernamen am System anmelden können, wenn Sie etwa `user1` und `user2` erlauben wollen, sich als `Administrator` anzumelden. Hier ergibt sich zwangsläufig, dass sich entweder `user1` auch als `user2` anmelden kann, oder umgekehrt. Dies lässt sich nur vermeiden, indem Sie eine solche Erlaubnis auf jeweils einen Benutzer beschränken und an evtl. weitere Benutzer zusätzliche Generatoren verteilen, für die der zugehörige (Dummy-)Benutzername dann im System nicht vorhanden ist.

  Beachten Sie auch die Besonderheiten zur Resynchronisation, die unter dem Punkt *Benutzer resynchronisieren* erklärt werden.

  Bemerkung:

Dieser Menüpunkt ist nur in der Administratoren-Ansicht aktiviert.

Gruppe:

- ***Gruppenverwaltung***
  Hiermit wird definiert welcher Gruppe ein Benutzer angehört bzw. für welche Gruppe ein Administrator verantwortlich ist.
  Sie erhalten ein Dialogfenstern in dem angezeigt wird, für welche Benutzer/Administratoren die Gruppenzugehörigkeit geändert werden soll. In der linken Auswahlspalte sehen Sie die Gruppen, die Sie einem Benutzer/Administrator zuweisen können, in der rechten die Schnittmenge der Gruppen der selektierten Einträge in der Datenbank. Durch die beiden Pfeittasten werden dann die Zuweisungen geändert.

  Warnung:
  Werden bei einem Benutzer alle Gruppen entfernt, so wird der User als undefiniert (undef) in der Datenbank geführt. Nur noch der Superadmin kann auf diesen Benutzer zugreifen bzw. ihm eine neue Gruppe zu weisen.
  Entfernt ein Administrator seine eigene Gruppenangehörigkeit, so wird die Verbindung zum SecOVID-Server unterbrochen. Der Administrator kann sich nicht mehr auf dem SecOVID-Server erfolgreich einloggen. Nur ein Superadmin kann dem Administrator wieder eine Gruppe zuweisen.

- ***Gruppenname***
  Lokale Zuordung eines Gruppennamens zu einer Gruppennummer. Durch einen Doppelklick oder einen „rechten Mausklick“ kann der Gruppenname ediert werden. Der zugeordnete Gruppenname wird nun in allen Dialogen bzw. Ansichten anstatt der Gruppennummer verwendet.

Hilfe:

- ***Info***
  Sie sehen ein Infofenster mit der Erreichbarkeit des Herstellers.

Die Menüleiste als Erweiterter Helpdesk

Die Menüleiste eines erweiterten Helpdesks enthält folgende Menüpunkte

- Programm
  - Konfiguration
  - Beenden
- Datenbank
  - Ansicht aktualisieren
  - Benutzer Information
  - Benutzer sperren
  - Benutzer entsperren
  - Benutzer suchen
  - Benutzer resynchronisieren
  - Server-PIN ändern
  - Administratoren-Datenbank

- Gruppe
  - Gruppenname
- Hilfe
  - Info

Die Menüleiste als Helpdesk

Die Menüleiste eines Helpdesks enthält folgende Menüpunkte

- Programm
  - Konfiguration
  - Beenden
- Datenbank
  - Ansicht aktualisieren
  - Benutzer Information
  - Benutzer sperren
  - Benutzer suchen
  - Benutzer resynchronisieren
  - Server-PIN ändern
  - Administratoren-Datenbank
- Gruppe
  - Gruppenname
- Hilfe
  - Info

3.1.2.5 Die Buttons

Die wichtigsten Funktionen der Software können Sie neben der Auswahl über die unter 1. beschriebene Menüleiste auch über Buttons aufrufen:

- ***Neuer Benutzer:***
  Anlegen neuer Benutzer (s.o.)
- ***Benutzer DB/Administrator DB:***
  Auswahlbox zum Hin- und Herschalten zwischen Benutzer- und Administratoren-Datenbank (s.o.)
- ***Karte erzeugen:***
  Generieren der zu den selektierten Benutzern gehörenden Passwortgeneratoren und Schreiben derselben auf Chipkarten (s.o.)
- ***SoftToken erzeugen:***
  Generieren von SoftToken für die selektierten Benutzern und Speichern derselben in entsprechenden Dateien ¡Benutzername¿`.kst` (s.o.)
- ***Benutzer sperren:***
  Sperren von Benutzern, so dass diesen der Zugang zum System verwehrt bleibt.

- ***Benutzer entsperren:***
  Entsperren von Benutzern, so dass diese mit korrektem Einmalpasswort wieder Einlass zum System erhalten.
- ***Ansicht aktualisieren:***
  Aktualisieren der angezeigten Daten durch Zugriff auf die Datenbank.
- ***Suchen:***
  Suchen nach Benutzern in der Datentabelle. Bei Anklicken dieses Buttons erscheint das bereits weiter oben beschriebene Fenster, in dem die Suchparameter angegeben werden können.

3.1.2.6 Sonstiges

Weitere Bedienungsmerkmale:

- **Anzeigen von Details zu einem Benutzer und Editieren der Einträge**
  Bewegen Sie den Mauszeiger auf den Benutzer, zu dem Sie detailliertere Informationen wünschen. Halten Sie die rechte Maustaste gedrückt und ziehen Sie den Mauszeiger auf den jetzt erscheinenden Balken „*Information*“ oder durch einen Doppelklick auf den Benutzer. Daraufhin sehen Sie folgende Informationen zum Benutzer:
  - den Benutzernamen *(editierbar)*
  - Name und Vorname *(editierbar)*
  - weitere Generatoren
  - die PIN,
    - die auf die Karte des Benutzers geschrieben wurde bzw. beim Anlegen eines Generators auf die Karte geschrieben wird *(editierbar)*
    - die bei der Verwendung eines Tokens mit PIN-Schutz eingegeben werden muss.
    - die als Server-PIN verwendet wird
  - die PUK, die auf die Karte geschrieben wurde bzw. beim Anlegen eines Generators auf die Karte geschrieben wird
  - die Seriennummer des Tokens bzw. der Chipkarte
  - der Generator, der vom Benutzer zum Erzeugen eines Einmalpasswort verwendeten wird.
  - die Gruppe(n) in der der Benutzer Mitglied ist
  - wann die Karte/Token erstellt wurde
  - wann der Benutzer sich beim SecOVID-Server zuletzt angemeldet hatte
  - das letzte Einmalpasswort des Benutzers
  - die SecOVID-Server-Systemzeit, zu welcher der ggf. für eine gewisse Zeitspanne gesperrte Benutzer sich wieder mit korrektem Einmalpasswort authentifizieren kann.
  - unter „Status“: ob der Benutzer gesperrt wurde oder nicht. (Im letzten Fall gibt z.B. „2 Fehler“ die Anzahl der Fehlversuche (hier: 2) seit dem letzten erfolgreichen Einloggen an.)
- **Alphabetisches Sortieren**
  Sie können den Inhalt Ihrer Datenbanktabelle nach allen Keys (Spaltennamen) sortieren lassen, die in Ihrem Programm angezeigt werden. Also z.B. nach Benutzer ID, Name, Vorname etc. Sie lassen dabei sortieren, indem Sie auf den Key klicken, nach dem alphabetisch sortiert werden soll. Verwenden Sie die linke Maustaste, wenn Sie aufsteigend sortieren wollen und die rechte Maustaste, wenn Sie absteigend sortieren wollen.

- **Selektieren mehrerer Benutzer**
  Wollen Sie eine Aktion für mehrere Benutzer durchführen, so markieren Sie die betreffenden Benutzereinträge in der Datentabelle und wählen den Menüpunkt zum Durchführen der gewünschten Aktion aus. Einen einzelnen Benutzer markieren Sie durch Anklicken des entsprechenden Benutzereintrages in der Datentabelle. Sind bereits Benutzer markiert, so können Sie mit einer der folgenden Methoden weitere Benutzer markieren:
  - Betätigen der linken Maustaste bei gedrückter „Strg-Taste“: Sie wählen *einen* weiteren Benutzer aus oder entfernen einen Benutzer aus der Auswahl.
  - Betätigen der linken Maustaste bei gedrückter „Shift-Taste“: Sie wählen auf einmal *eine Serie* von weiteren Benutzern aus.
  - Ziehen Sie die Maus bei gedrückter linker Maustaste über Benutzereinträge: Sie wählen auf einmal *eine Serie* von Benutzern aus, wobei alle anderen Benutzer aus der Auswahl entfernt werden. Falls Sie während dieser Operation die „Strg-Taste“ gedrückt halten, werden die Benutzer zu bisherigen Auswahl hinzugefügt.
  - Benutzen Sie die Suchfunktion. Die bei der Suche gefundenen Benutzer sind nun zusätzlich zu den bereits selektierten Benutzern markiert.
- **Selektion von Benutzern aufheben**
  Sie können die Selektion eines Benutzers aufheben, indem Sie den betreffenden Benutzereintrag in der Datentabelle bei gedrückter „Strg-Taste“ mit der linken Maustaste anklicken. Die Selektion aller markierten Benutzer heben Sie auf, indem Sie die „Esc-Taste“ drücken. Alternativ können Sie zunächst mit der linken Maustaste auf einen beliebigen Benutzereintrag in der Datentabelle klicken (nun ist nur dieser Benutzer selektiert) und anschließend die Selektion dieses Benutzers wie oben beschrieben aufheben.

3.1.2.7 Lokale Richtlinien für Benutzernamen

Um dafür zu sorgen, dass alle Benutzernamen die von den Administrationstools oder der Administrations-Bibliothek erzeugt werden, Ihren lokalen Richtlinien genügen, können Sie eine Bibliothek mit dem Namen `libSecSyntaxCheck.so` (Unix) bzw. `SecSyntaxCheck.dll` (Windows) erzeugen. Diese Bibliothek wird auf dem System neben der Administrations-Bibliothek abgelegt, d.h. typischerweise im selben Verzeichnis wir die Adminitrationstools selbst. Sie muss die Funktion `int FilterUsername(char *)` (mit der Standard-C-Calling-Konvention) zur Verfügung stellen. Dieser Funktion wird ein Benutzername übergeben und sie soll 0 zurückgeben, wenn der übergebene Benutzername bereits gültig ist, sie soll den Benutzernamen automatisch der lokalen Richtlinie anpassen und einen positiven Wert zurückgeben, wenn dies möglich ist, und sie soll einen negativen Wert zurückgeben, wenn der Benutzername ungültig ist und nicht automatisch angepasst werden kann.
Der C-Code für ein einfaches Beispiel einer solchen Funktion könnte im Kern so aussehen:

```
#include <ctype.h>
#include <string.h>
#include <stdlib.h>
int FilterUsername(char *name)
{
  int i, ret = 0;
  char * lname = strdup(name);
  for (i= 0; i < strlen(name); i++)
    lname[i] = tolower(lname[i]);
  if (strcmp(name, lname)){
    ret = 1;
    strcpy(name, lname);
  }
  free(lname);
  return ret;
}
```

Diese Funktion wandelt jedes Zeichen des Benutzernamens automatisch in den entsprechenden Kleinbuchstaben um und gibt am Ende entweder `1` oder `0` zurück, je nachdem, ob sich eine Änderung ergab oder nicht.

3.1.3 Kommandozeilen-Administrationstool

Neben dem graphischen Administrationstool gibt es auch ein Kommandozeilentool zur Verwaltung von SecOVID-Benutzern, SecOVID-Tokens und SecOVID-Chipkarten. Der Einsatz des Kommandozeilentools macht in folgenden Szenarien Sinn:

- Sollen neben der SecOVID-Applikation auf den Chipkarten der Benutzer weitere Applikationen angelegt werden, so wünscht sich der Betreiber des Gesamtsystems ein einziges Administrationstool (ggf. mit GUI) für alle Applikationen. Durch das Kommandozeilentool können einfach die spezifischen Kommandos und Daten an den SecOVID-Server (bzw. die SecOVID-Datenbank) übermittelt werden. Somit kann das Kommandozeilentool leicht in die GUI zur Verwaltung aller Chipkartenapplikationen integriert werden.
- Wird der SecOVID-Server auf einem dedizierten Server in einem sicheren Umfeld installiert, den man im wesentlichen von anderen Rechnern aus administrieren will, dann wird auf diesem Rechner häufig *nicht* die graphische Oberfläche X installiert sein. Dann hat man aber Probleme, eine initiale Administratorkarte zu erzeugen bzw. im Falle des Verlusts eine neue Administratorkarte auszustellen. Mit dem Kommandozeilentool läßt sich nun ein neuer Administrator einfach mit dem Kommando

  ```
  /etc/SecOvid/admin/SecAdm -new_user adminlogin=Administrator pin=123456
  ```

  erzeugen (optional können Sie noch Nach- und Vornamen angeben). Eine neue Karte kann anschließend mit

  ```
  /etc/SecOvid/admin/SecAdm -generate_card adminlogin=Administrator
  ```

  erzeugt werden.

Hinweis: Bei verschiedenen Aktionen mit dem Kommandozeilentool (z.B. zur Remote-Administration des SecOVID-Servers) müssen die Treiber für das angeschlossene KOBIL-Chipkartenterminal gefunden werden. Unter UNIX-Systemen müssen Sie hierzu die Umgebungsvariable `LD_LIBRARY_PATH` korrekt setzen. Alternativ können Sie hier auch statt `SecAdm` das Script `SecAdm.sh` aufrufen, welches diese Aufgabe übernimmt. Auf Windows-Systemen wird erwartet, dass der Treiber in einem Verzeichnis gefunden werden kann, das in einem der Verzeichnisse der Umgebungsvariable `PATH` enthalten ist.

Vorbereitungen zum Starten des SecOVID-Kommandozeilentools Beachten Sie, dass das rein textbasierte SecOVID-Kommandozeilentool nur Benutzer und Karten verwalten kann, falls der SecOVID-Server in Betrieb ist. Eine Remote-Administration ist nur möglich, wenn Sie bereits über eine am SecOVID-Server registrierte SecOVID-Administratorchipkarte verfügen. Zur erfolgreichen *Remote*-Administration des SecOVID-Servers stellen Sie zudem sicher, dass Ihre Firewalls und sonstigen Paketfilter die TCP-Kommunikation zwischen SecOVID-Admintool (Kommandozeilentool) und SecOVID-Server nicht behindern. Das bedeutet, die Firewall muss alle TCP-Pakete durchlassen, die vom SecOVID-Kommandozeilentool zum SecOVID-Server auf dessen Datenbankport (in der Regel Port 1113, siehe den Eintrag `secoviddb` in der Datei `/etc/services` des SecOVID-Servers) gesendet wird. Zudem muss die Firewall alle TCP-Pakete des SecOVID-Servers an das SecOVID-Kommandozeilentool (undefinierter Port) durchlassen.

Die Konfigurationsdatei `Prefs.cfg`

Zudem müssen Sie vor dem Starten des Tools sicherstellen, dass im lokalen Verzeichnis eine Konfigurationsdatei `Prefs.cfg` existiert. Diese könnte folgenden Inhalt haben:

```
localhost
1113
1
0
```

```
0
0
6
6
0
0
```

Die Datei `Prefs.cfg` besteht aus mindestens zehn Zeilen (und höchstens 14 Zeilen), deren Inhalt Sie gemäß Ihrem System Zeile für Zeile anpassen müssen:

Zeile 1: Enthält die IP-Adresse des SecOVID-Servers. In obigem Beispiel ist der SecOVID-Serverprozess also auf dem lokalen Rechner installiert.

Zeile 2: Enthält die Portnummer, an der der SecOVID-Server Administrationsanfragen entgegennimmt. Tragen Sie die Portnummer ein, welche in der Datei `/etc/services` auf dem Rechner des SecOVID-Servers dem Interface `secoviddb` zugeordnet ist. Der Standard-Port für Administrationsanfragen lautet 1113.

Zeile 3: Enthält den COM-Port, an dem das KOBIL-Chipkartenterminal angeschlossen ist. In obigem Beispiel ist COM-Port 1 angegeben.

Zeile 4: Der Wert `0` bedeutet, dass beim Starten des Kommandozeilentools nicht automatisch nach einem Chipkartenterminal gesucht wird.
Der Wert `1` bedeutet, dass beim Starten des Kommandozeilentools automatisch nach einem Chipkartenterminal gesucht wird.
Andere Werte als `0` oder `1` sind nicht zulässig. In obigem Beispiel wird nicht automatisch nach einem Chipkartenterminal gesucht, d.h. im jeweiligen Kommando müsste explizit angegeben werden, an welchem COM-Port das Chipkartenterminal installiert ist.

Zeile 5: Der Wert `0` bedeutet, dass die (Chipkarten-)PIN bei neu anzulegenden Benutzern rein-numerisch sein soll (nur Ziffern).
Der Wert `1` bedeutet, dass die (Chipkarten-)PIN bei neu anzulegenden Benutzern alphanumerisch sein soll (Buchstaben und Ziffern).
Andere Werte als `0` oder `1` sind nicht zulässig. In obigem Beispiel bestehen künftig zu verwendende (Chipkarten-)PINs nur aus Ziffern.

Zeile 6: Der Wert `0` bedeutet, dass die (Chipkarten-)PUK bei neu anzulegenden Benutzern rein-numerisch sein soll (nur Ziffern).
Der Wert `1` bedeutet, dass die (Chipkarten-)PUK bei neu anzulegenden Benutzern alphanumerisch sein soll (Buchstaben und Ziffern).
Andere Werte als `0` oder `1` sind nicht zulässig. In obigem Beispiel bestehen künftig zu verwendende (Chipkarten-)PUKs nur aus Ziffern.

Zeile 7: Diese Zeile enthält die Länge der (Chipkarten-)PIN für neu anzulegenden Benutzer. Im Beispiel sind (Chipkarten-)PINs für neu anzulegende Benutzer sechsstellig.

Zeile 8: Diese Zeile enthält die Länge der (Chipkarten-)PUK für neu anzulegende Benutzer. Im Beispiel sind (Chipkarten-)PUKs für neu anzulegende Benutzer sechsstellig.

Zeile 9: Es können bis zu 10 verschiedene Einmalpaßwortgeneratoren auf eine Chipkarte geschrieben werden (Paßwortgenerator Nr. 0 bis Paßwortgenerator Nr. 9). Der Wert `1` in Zeile 9 bedeutet, dass beim Schreiben (oder Löschen) eines Einmalpaßwortgenerators auf eine Chipkarte immer der in Zeile 10 spezifizierte Default-Generator verwendet werden soll. (Der tatsächlich zu verwendende Paßwortgenerator kann in der eigentlichen Anweisung zum Schreiben oder Löschen auf der Chipkarte nochmals angepaßt werden.)
Der Wert `0` bedeutet, dass beim Schreiben oder Löschen eines Einmalpaßwortgenerators auf der Chipkarte kein Default-Generator benutzt werden soll.
Andere Werte als `0` oder `1` sind nicht zulässig. In obigem Beispiel ist kein Default-Generator gesetzt.

Zeile 10: Diese Zeile enthält den zu verwendenden Einmalpaßwortgenerator (Nr. 0, ..., Nr. 9). Dieser Eintrag besitzt nur Relevanz, wenn in Zeile 9 durch den Wert `1` codiert ist, dass ein Default-Generator verwendet werden soll und wenn in der Anweisung zum Schreiben oder Löschen nicht explizit ein Paßwortgenerator angegeben ist. In obigem Beispiel würde Paßwortgenerator Nr. 0 als Default-Wert definiert werden, wenn in Zeile 9 die Benutzung eines Default-Wertes aktiviert wäre.

Zeile 11: (optional, d.h. diese Zeile können Sie zunächst leer lassen.) Diese Zeile enthält den Default-Pfad, der beim Exportieren von Benutzereinträgen verwendet wird. Der Pfad zur Export-Datei kann bei jedem Exportvorgang nochmals angepaßt werden. Der Default-Eintrag ist `./`, also das aktuelle Verzeichnis.

Zeile 12: (optional, d.h. diese Zeile können Sie zunächst leer lassen.) Diese Zeile enthält den Default-Pfad, der beim Importieren von Benutzereinträgen verwendet wird. Der Pfad zur Import-Datei kann bei jedem Importvorgang nochmals angepaßt werden. Der Default-Eintrag ist `./`, also das aktuelle Verzeichnis.

Zeile 13: (optional, d.h. diese Zeile können Sie zunächst leer lassen.) Diese Zeile enthält den Default-Pfad zur Log-Datei, die Probleme beim Importieren und Exportieren von Benutzereinträgen festhält. Der Pfad zur Log-Datei kann bei jedem Importvorgang nochmals angepaßt werden. Der Default-Eintrag ist `./OvidLog.log`.

Zeile 14: (optional, d.h. diese Zeile können Sie zunächst leer lassen.) Diese Zeile enthält den Default-SecOVID-Administratornamen. Der Eintrag hat nur Relevanz, wenn Sie im Kommandozeilentool nicht explizit den Administratornamen mit übergeben.

Hinweis: Die Datei `Prefs.cfg` können Sie auch mit dem SecOVID-Administrationstool (GUI) erzeugen. Starten Sie hierzu das SecOVID-Administrationstool (GUI) und nehmen Sie die gewünschten Einstellungen über den Menüpunkt Programm > Konfiguration vor. Die Datei `Prefs.cfg` wird dann im lokalen Verzeichnis auf der Festplatte erzeugt.

Die Gruppendatei `groups.cfg`
In der Gruppendatei `groups.cfg` können den Gruppennummern Gruppennamen zugeordnet werden. Wird von dieser Möglichkeit kein Gebrauch gemacht (weil die Datei `groups.cfg` nicht existiert oder weil für eine Gruppennummer nicht explizit ein Name definiert wurde), so ist die Gruppennummer auch gleichzeitig der Gruppenname. Es existieren insgesamt 63 Gruppen; diese haben die Gruppennummern 1 bis 63. In der Datei `groups.cfg` gibt es für jede dieser Gruppen eine Zeile, in der Gruppennummer und Gruppenname durch einen Doppelpunkt getrennt sind.
Die Datei `groups.cfg` könnte folgenden Inhalt haben:

```
1:Worms
2:München
3:Frankfurt
...
62:Stuttgart
63:Dortmund
```

Hinweis: Die Datei `groups.cfg` können Sie auch mit dem SecOVID-Administrationstool (GUI) erzeugen. Starten Sie hierzu das SecOVID-Administrationstool (GUI) und nehmen Sie die gewünschten Einstellungen über den Menüpunkt Gruppe > Gruppenname vor. Die Datei `groups.cfg` wird dann im lokalen Verzeichnis auf der Festplatte erzeugt.

3.1.3.1 Benutzung des SecOVID-Kommandozeilentools

Die allgemeine Syntax sieht wie folgt aus: **SecAdm** [**Optionen** [**Kommando Parameterliste**]]

BESCHREIBUNG
SecAdm weist nahezu die komplette Funktionalität des SecOVID-Administrationstools mit graphischer Oberfläche auf. Das Programm läßt sich von der Kommandozeile aus mit Optionen und Parametern starten, welche die gewünschte Aktion festlegen. Das Programm liefert bei jeder Aktion Rückgabewerte zurück, die den Erfolg

oder Mißerfolg einer Aktion anzeigen. Standardinput und Standardoutput können durch die entsprechenden Operatoren umgeleitet werden.
Hinweis: Beachten Sie, dass je nach verwendeter Shell bestimmte Zeichen eine spezielle Bedeutung haben und von der Shell interpretiert werden. Für die Übergabe an das Administrationstool durch den Shell-üblichen Mechanismus (z.B. durch einen vorangestellten Backslash oder durch Verwendung von Anführungszeichen) vor der Verarbeitung durch die Shell geschützt werden müssen.

OPTIONEN

[-admin *string*] [-admin_port *N*]
Diese Optionen können allen Kommandos vorangestellt werden und weisen das Administrationstool an, sich mit dem angegebenen Namen an der Datenbank anzumelden, dabei wird zur Authentifikation das Chipkartenterminal an der seriellen Schnittstelle mit der angegebenen Nummer verwendet. Ist keine spezifische Schnittstelle angegeben, so wird dieselbe Schnittstelle wie für die anderen Kartenoperationen (s.u.) verwendet, wobei gegebenenfalls eine Aufforderung zum Wechseln der Chipkarte erscheint. Wird kein Administrator angegeben, so wird der in der Datei Prefs.cfg, die das graphische Administrationstool anlegt, gespeicherte Administratorname verwendet.

[-admin *string*] [-admin_token *Dateiname:PIN*]
Alternativ können diese Optionen allen Kommandos vorangestellt werden. Im Gegensatz zur vorangegangenen Variante wird hier keine Chipkarte, sondern ein zuvor erzeugtes „Softtoken“ zur Authentifikation gegenüber der Datenbank benutzt. Dabei gibt *Dateiname* den Namen der Datei (am besten mit absolutem Pfad) an, die die Daten des Softtokens beinhaltet, und PIN gibt den Schlüssel an, mit dem diese Datei entschlüsselt werden kann. Beachten Sie, dass diese PIN ein hinreichend gutes statisches Passwort sein sollte (also zumindest 8 Zeichen, mindestens ein Sonderzeichen, und kein Wort, dass sich in einem Wörterbuch finden läßt) sein sollte, da jeder, der dieses Passwort (im Laufe beliebig vieler Versuche) erraten kann, Zugriff auf Ihre Datenbank erhält. Wir empfehlen, diese Option höchstens für „Helpdesk“-Administratoren zu verwenden. Beachten Sie, dass der angegebene Administratorname zu den Daten in der angegebenen Datei passen muss, ansonsten kann keine Verbindung zur Datenbank hergestellt werden. Wird kein Administrator angegeben, so wird der in der Datei Prefs.cfg, die das graphische Administrationstool anlegt, gespeicherte Administratorname verwendet.

PARAMETER

[adminlogin] oder [admin]
Durch Angabe von **adminlogin** anstelle von **login** können sie angeben, daß ein bestimmtes Kommando für einen Administrator und nicht für einen normalen Benutzer ausgeführt werden soll. Bei den Optionen, die keinen Parameter **login** haben erreichen sie den gleichen Effekt durch die Angabe des zusätzlichen Parameters **admin**.

KOMMANDOS

Anzeige, Sortieren und Suchen

-show [admin]
Zeigt alle User bzw. alle Administratoren an.
Die Integervalues der Spalte "status" haben folgende Bedeutung:

- -1
  Der Benutzer hat den Status "Gesperrt"
- 0
  Der Benutzer hat den Status "OK"
- 1-9
  Der Benutzer hat n mal ein falsches OTP benutzt
- -2 und -3
  Der Administrator hat für den Benutzer die Resynchronisation eingeleitet. Da bei der Resynchronisation zwei OTPs angefordert werden, ist dieses ein zweistufiger Vorgang, wobei die erste Stufe durch -2, die zweite durch -3 angezeigt wird.

-show [admin]login=*stringlist1* [lastname=*stringlist2*] [firstname=*stringlist3*] [group=*stringlist4*]
Zeigt alle User an, die mit den angegebenen Suchkriterien gefunden werden. Die unterschiedlichen Suchkriterien werden durch ein logisches "ODER" verknüpft.

Neuer Benutzer
-new_user [admin]login=*string1* [lastname=*string2*] [firstname= *string3*] [pin=*string4*] [group=*stringlist*]
Legt einen neuen User oder Administrator mit den angegebenen Daten an. Dabei ist die Angabe des Parameters **login** notwendig. Die anderen Parameter sind optional. Wenn keine **pin** angegeben ist, wird automatisch eine zufällige PIN generiert. Wenn keine **group** angegeben ist, wird der Benutzer Mitglied der Gruppe(n), die von diesem Administrator verwaltet werden. Legt ein Superadmin einen User ohne diesen Gruppenparameter an, wird der Benutzer als **"Undefined"** gekennzeichnet. Beachten Sie, dass in keiner der eingegebenen Zeichenfolgen, also weder im Benutzernamen, noch im Namen oder Vornamen oder in der PIN, Doppelpunkte vorkommen dürfen. Darüber hinaus sind im Benutzernamen auch Leerzeichen sowie die beiden Zeichen `@` und `/` verboten.
-new_user_batch [admin]login=*string* startno=*N1* endno=*N2* [pin=*string4*] [group=*stringlist*]
Legt neue User oder Administratoren automatisch im Batchbetrieb mit den angegebenen Daten an. Beispiel:
SecOvid_adm -new_user_batch login=guest startno=1 endno=1000 pin=421673 group=1,5
legt automatisch 1000 Benutzer mit den Benutzernamen **guest0001** bis **guest1000** und der PIN **421673** an. Alle angelegten Benutzer sind Mitglied der Gruppen **1** und **5**.
Wenn keine **pin** angegeben ist, wird automatisch eine zufällige PIN für jeden Benutzer generiert. Wenn keine **group** angegeben ist, wird der Benutzer Mitglied der Gruppe(n), die von diesem Administrator verwaltet werden.

Benutzerdaten ändern
-update login=*string1* [new-login=*string2*] [new-lastname=*string3*] [new-firstname=*string4*] [new-pin=*string5*] [new-group=*stringlist*] [new-gentype=*string6*]
Ändert Login, Nachnamen, Vornamen, PIN, Gruppe und/oder Generatortyp eines Users. Beachten Sie wiederum, dass in keiner der eingegebenen Zeichenfolgen, also weder im Benutzernamen, noch im Namen oder Vornamen oder in der PIN, Doppelpunkte vorkommen dürfen. Darüber hinaus sind im Benutzernamen auch Leerzeichen sowie die beiden Zeichen `@` und `/` verboten.
Wenn Sie die PIN ändern beachten Sie bitte die Hinweise im Abschnitt *PIN/PUK* von Kapitel 3.1.2.4.
Mögliche Gruppeneinträge:

- *undef* – Undefiniert, der Benutzer gehört keiner Gruppe an
- *Gruppenname1[, Gruppenname2]*

Mögliche Generatorentypen:

- *Undefined* – Undefiniert
- *Unknown* – Unbekannt
- *Chipcard* – Chipkarte
- *Chipcard-6-digit* – Chipkarte für 6-stellige Einmalpasswörter
- *Token/PIN* – SecOVID Token mit PIN-Schutz (nicht mehr unterstützt)
- *Token-6-digit/PIN* – SecOVID Token für 6-stellige Einmalpasswörter mit PIN-Schutz (nicht mehr unterstützt)
- *Token/SP* – SecOVID Token mit Server-PIN
- *Token-6-digit/SP* – SecOVID Token für 6-stellige Einmalpasswörter mit Server-PIN
- *Token/PIN/SP* – SecOVID Token mit PIN-Schutz und Server-PIN (nicht mehr unterstützt)

- *Token-6-digit/PIN/SP* – SecOVID Token für 6-stellige Einmalpasswörter mit PIN-Schutz und Server-PIN (nicht mehr unterstützt)
- *Token* – SecOVID Token
- *Token-6-digit* – SecOVID Token für 6-stellige Einmalpasswörter
- *SoftToken* – SecOVID SoftToken
- *SoftToken/SP* – SecOVID SoftToken mit Server-PIN
- *Static* – Statisches Notfallpasswort
- *Alias* – Reiner Alias ohne eigenen Generator

Bei der Migration eines RSA ACE/Server muss dem Generatortyp eine zusätzliche Option "*/ace*" angefügt werden. Dieses Flag signalisiert dem SecOVID-Server, dass die Benutzerauthentifizierung durch den RSA ACE/Server vorgenommen wird.

Administratordaten ändern
-update adminlogin=*string1* [**new-login=*string2***] [**new-lastname=*string3***] [**new-firstname=*string4***] [**new-group=*stringlist***] [**new-admintype=*string5***]
Ändert Login, Nachnamen, Vornamen, Gruppe und/oder Administratortyp eines Administrators.
Mögliche Gruppeneinträge:

- *all* – Administrator aller Gruppen ("Superadmin")
- *Gruppenname1[, Gruppenname2]*

Mögliche Adminstratorentypen:

- *admin* – Administrator
- *adv.helpdesk* – erweiterter Helpdesk
- *helpdesk* – Helpdesk

Benutzer sperren/ entsperren
-lock login=*stringlist*
-unlock login=*stringlist*
Sperrt bzw. entsperrt Benutzer in der SecOVID-Datenbank.

Benutzer resynchronisieren
-resync login=*stringlist*
Merkt Benutzer in der SecOVID-Datenbank zur Resynchronisation vor.

Benutzer löschen
-delete [admin]login=*stringlist*
Löscht Benutzer oder Administratoren aus der SecOVID-Datenbank.

Import und Export von Benutzern
-import_db [admin] *filename*
-export_db [admin] *filename*
Importiert bzw. exportiert Benutzerdaten oder Administratordaten aus der SecOVID-Datenbank.

Karte erzeugen
-generate_card [admin]login=*string* [genno=*N*]
-generate_card6 login=*string* [genno=*N*]
Für den angegebenen Benutzer oder Administrator wird eine Karte erzeugt. Durch Verwendung von **generate_card6** wird dabei festgelegt, dass ein Generator zur Erzeugung 6-stelliger Einmalpasswörter erzeugt werden soll. Der Parameter **genno=*N***spezifiziert die Nummer des Paßwortgenerators, der angelegt werden soll (***N* ist eine der Zahlen 0,1, ..., 9). Fehlt dieser Parameter, so wird die im Konfigurationsfile angegebene Nummer als Default benutzt. Für Administratoren wird immer der Generator Nummer 9 verwendet.
Bitte beachten Sie auch die Hinweise zum Punkt *Karte erzeugen* in Kapitel 3.1.2.4.

Applikation löschen
-delete_card [admin]login=*string*
Löscht die gesamte SecOVID-Applikation des angegebenen Benutzers oder Administrators von dessen Chipkarte (alle Paßwortgeneratoren, PIN, PUK).

Paßwortgenerator löschen
-delete_generator [admin]login=*stringlist* [genno=*N*]
Löscht den angegebenen Paßwortgenerator (und sonst nichts) auf der Chipkarte des angegebenen Benutzers. Ist kein Paßwortgenerator explizit angegeben, so wird der Default-Generator gelöscht (siehe Konfigurationsfile). Für Administratoren wird immer der Schlüssel zur Anmeldung als Administrator gelöscht.

PIN zurücksetzen
-unblock_pin [admin]login=*stringlist*
Setzt die PIN auf der Chipkarte des angegebenen Benutzers oder Administrators auf den Wert zurück, der für den Benutzer bzw. Administrator in der SecOVID-Datenbank gespeichert ist.

SoftToken für Administrator erzeugen
-generate_softtoken adminlogin=*string*
Für den angegebenen Administrator wird ein SoftToken erzeugt. Dieses Softtoken wird in der Datei *string*.kst gespeichert. Beachten Sie, dass die PIN des angegebenen Administrators, die in der Datenbank gespeichert ist, benutzt wird, um das File zu verschlüsseln. Aus Sicherheitsgründen sollten sie unbedingt sicherstellen, dass diese PIN ein ausreichend gutes Passwort ist, z.B. indem Sie es mit Hilfe des **-update** Kommandos ändern, bevor Sie das **-generate_softtoken** Kommando ausführen. Wie oben erwähnt sollte das Passwort 8 Zeichen lang sein und kein Wort sein, dass in einem Wörterbuch zu finden ist.

SoftToken für Benutzer erzeugen
-generate_softtoken login=*string1* [gentype=*string2*] [add]
Für den angegebenen Benutzer wird ein SoftToken erzeugt. Dieses Softtoken wird in der Datei *string1*.kst gespeichert. Beachten Sie, dass die PIN des angegebenen Benutzers, die in der Datenbank gespeichert ist, benutzt wird, um das File zu verschlüsseln. Aus Sicherheitsgründen sollten sie sicherstellen, dass diese PIN ein ausreichend gutes Passwort ist, z.B. indem Sie es mit Hilfe des **-update** Kommandos ändern, bevor Sie das **-generate_softtoken** Kommando ausführen. Wie oben erwähnt sollte das Passwort 8 Zeichen lang sein und kein Wort sein, dass in einem Wörterbuch zu finden ist.
Als **gentype** können Sie entweder „SoftToken“ oder „SoftToken/SP“ angeben, je nachdem ob sie zusätzlich eine Server-PIN verwenden wollen oder nicht, defaultmäßig wird das SoftToken ohne Server-PIN benutzt.
Für einen Web-Server zum Download von SoftToken wird zusätzlich eine Datei `access.dat` generiert, in denen Benutzernamen und die PINs dieser Benutzer als Zugriffsparameter für die Benutzerauthentifikation gespeichert werden. Mit dem Parameter **add** können sie festlegen, dass der neue Eintrag für das gerade generierte SoftToken dieser Datei hinzugefügt werden soll, ohne diesen Parameter erfolgt eine Fehlermeldung, falls die Datei bereits existiert und es wird kein SoftToken erzeugt.

Notfall-Passwort setzen
-generate_static login=*string1* [password=*string2*] [days=*N*]
Für den angegebenen Benutzer wird ein statisches Notfall-Passwort erzeugt. Diese Funktion kann z.B. dazu dienen, einem Benutzer, der seinen Einmalpasswortgenerator zu Hause vergessen hat, dennoch den Zugriff zu ermöglichen. Dabei wird eine eventuell gesetzte Server-PIN mit diesem Notfall-Passwort überschrieben, so dass der Benutzer nach Ablauf des Gültigkeitszeitraums ggf. seine Server-PIN wieder neu setzen muss.
Mit dem Parameter **password** kann explizit das Notfall-Passwort übergeben werden, defaultmäßig wird ein Passwort aus 8 zufälligen alphanumerischen Zeichen generiert und ausgegeben.
Mit dem Parameter **days** kann spezifiziert werden, wie lange das Passwort gültig sein soll, DefaultWert ist 1 Tag. Nach Ablauf der angegebenen Anzahl von Tagen gelten automatisch nur noch die Einmalpasswörter des bisherigen Generators, ggf. in Verbindung mit dem Notfall-Passwort als neuer Server-PIN. Diese Server-PIN kann natürlich wieder in der üblichen Weise geändert werden.
Beachten Sie, dass das Notfall-Passwort ebenfalls automatisch ungültig wird, sobald der Benutzer erstmals wieder ein korrektes Einmalpasswort (ggf. mit dem Notfall-Passwort als neuer Server-PIN) verwendet.

Alias-Link setzen
-alias login=*string1* referred=[*string2*]
Dem angegebenen Benutzer wird es ermöglicht, auch den Generator des *string2* bezeichneten Benutzers zu benutzen, um sich zu authentifizieren. Beachten Sie, dass jeder Benutzer nur einen solchen Alias-Link haben kann, wenn Sie den Befehl ein zweites Mal verwenden, wird der schon vorhandene Link überschrieben. Insbesondere können sich durch Weglassen von *string2* erreichen, dass ein vorhandener Link wieder gelöscht wird.
Beachten Sie, dass es möglich ist, eine Kette von Links zu bilden, dass dann jedoch nicht nur alle Generatoren von Benutzern in dieser Kette für den ersten Benutzer verwendet werden können, sondern jeder Generator für *alle* Benutzer weiter vorne in der Kette gültig ist, Dies ist kein Problem, wenn sie Dummy-Benutzer verwenden, die es im System nicht gibt, um es einem Benutzer zu ermöglichen, mehrere verschiedene Generatoren zur Anmeldung zu benutzen. Es kann jedoch zum Problem werden, wenn Sie es ermöglichen wollen, dass sich ein Benutzer mit demselben Token unter verschiedenen Benutzernamen anmelden kann (z.B. zur Verwaltung von System-Accounts wie „Administrator“).

LDAP-Benutzer importieren
-add-ldap *Server-IP LDAP-Port Suffix Filter* [*LDAP-User* [*LDAP-Password*]]
Von dem LDAP-Server mit IP-Adresse *Server-IP*, der an Port *LDAP-Port* auf LDAP-Anfragen beantwortet, werden Benutzer gemäß *Suffix* und *Filter* importiert. Dabei werden *LDAP-User* und *LDAP-Password* benutzt, um sich gegenüber dem LDAP-Server zu authentifizieren. Für die detaillierte Erläuterung der einzelnen Parameter verweisen wir auf den Abschnitt *Datenbank importieren (LDAP):* in Kapitel 3.1.2.4.
Beachten Sie insbesondere, dass die dort beschriebene Datei `ldap.conf` existieren muss.

KONFIGURATION
Die Konfiguration (COM-Port für den Kartenleser, IP-Adresse und Port der SecOVID-Datenbank, ...) ist im Konfigurationsfile **Prefs.cfg** im lokalen Verzeichnis abgelegt. Diese kann manuell oder mit Hilfe des graphischen Verwaltungstools editiert werden.

RETURNCODES
Das Programm liefert bei jedem Aufruf einen Wert (Returncode) zurück, der Auskunft über den Ausgang der veranlaßten Aktion gibt. Der Returncode 0 hat die Bedeutung "Aktion erfolgreich ausgeführt." Alle anderen Returncodes (-1, -2, ...) zeigen den Mißerfolg der Aktion an und erläutern den Fehler genauer. Die Bedeutung des Returncodes wird als textuelle Fehlermeldung nach stdout ausgegeben. Es folgt die Liste aller möglichen Returncodes:

- Aktion erfolgreich ausgeführt. // 0
- Keine Verbindung zur Datenbank. // -1
- Datenbank-Server antwortet nicht oder User existiert nicht. // -2

- Löschen eines Benutzers fehlgeschlagen. // -3
- Benutzer nicht gesperrt. // -4
- Konnte nicht auf die Karte zugreifen: keine oder falsche Karte. // -5
- PIN-Verifikation fehlgeschlagen oder PIN zu kurz. // -6
- Aktualisieren der Datenbank fehlgeschlagen bzw. bei **-new_user_batch** auch zu kurze oder zu lange PIN. // -7
- Benutzer nicht vorhanden. // -8
- Benutzer bereits vorhanden. // -9
- Ungültige Parameter. // -10
- Ungültige Anzahl von Parametern. // 10
- Zu importierende Datenbankdatei nicht gefunden. // -11
- File mit PINs für Softtoken (siehe Abschnitt 3.1.7.1) konnte nicht geschrieben werden. // -12
- Befehl verweigert. Keine Berechtigung // -13
- Befehl verweigert. Keine Berechtigung den Benutzer in alle gewählten Gruppen aufzunehmen bzw. aus allen nicht gewählten Gruppen zu entfernen. // -14
- Angegebene Gruppe existiert nicht bzw. bei **-delete_card** auch: PIN konnte nicht zurückgesetzt werden bzw. bei **-add-ldap**: Anfrage an den LDAP-Server fehlgeschlagen. // -15
- Aktualisierung der Datei mit den PINS für Softtoken fehlgeschlagen. // -16
- Erzeugen des Softtokens fehlgeschlagen. // -17
- Falscher Admintype. // -18
- Unzulässige Verwendung von **new-gentype** bzw. **new-admintype**. // -19
- Benutzer kann nicht entsperrt werden, da noch keinen gentyp bzw. bei **-add-ldap**: Konfigurationsfile `ldap.conf` nicht gefunden. // -20
- Die Operation ist für diesen Benutzertyp nicht zulässig (z.B. Kartenoperation können nur für Chipkarte und nicht für Softtoken ausgeführt werden). // -21
- Der referenzierte Alias-Benutzer existiert nicht. // -22
- Der anzulegende Generator existiert bereits auf der Chipkarte. // -24
- Softtoken kann bei der Anmeldung nicht entschlüsselt werden. // -25
- Operation aus Sicherheitsgründen verweigert (nur bei **-generate_softtoken**). // -26
- Keine Chipkarte zur Zufallszahlenerzeugung gefunden (nur bei **-generate_softtoken**). // -27
- Interner Programmfehler. // -30, -31, -40
- Ungültiger Benutzername. // -35
- Ungültiger Nachname. // -36
- Ungültiger Vorname. // -37
- Ungültige PIN. // -38

Speziell beim Datenbank-Import oder -Export haben einige Fehlercodes derzeit eine andere Bedeutung, diese Sonderfälle sind in der folgenden Liste aufgezählt:

- Datenbankdatei kann nicht geöffnet werden oder hat ungültiges Format. // -1
- Schlüsselgenerierung für den Export fehlgeschlagen. // -2
- Datenbankdatei für den Import wurde von anderem Administrator angelegt. // -3
- Datenbank-Import hatte ein Problem, OvidLog.log enthält Details. // -6
- Datenbank-Export hatte ein Problem, OvidLog.log enthält Details. // -8, -16, -24

3.1.4 Administrations-Bibliothek

Neben den fertigen Administrationstools gibt es auch eine Bibliothek, mit deren Hilfe die SecOVID-Administration in eigene Programme und Oberflächen eingebunden werden kann. Diese Bibliothek hat ein C- und C++-Interface. Der zugehörige Header findet sich auf der CD als
`/Development/admin/admininterface.h`, die Bibliothek selbst wird unter dem Namen `libSecAdm.so` (Unix) bzw. `SecAdm.dll` (Windows) zusammen mit den Administrationstools installiert.
Beachten Sie das auch diese Bibliothek, ebenso wie die Administrationstools die Bibliothek für die lokalen Richtlinien bezüglich der Benutzernamen unterstützt.
Hinweis: Für verschiedenen Aktionen der Bibliothek (z.B. zur Personalisierung von Chipkarten) müssen die Treiber für das angeschlossene KOBIL-Chipkartenterminal gefunden werden. Unter UNIX-Systemen müssen Sie hierzu die Umgebungsvariable `LD_LIBRARY_PATH` korrekt setzen oder die Datei `libct.so` in eines der Verzeichnisse kopieren, dass per Default durchsucht wird, z.B. `/usr/lib`. Auf Windows-Systemen wird erwartet, dass der Treiber in einem Verzeichnis gefunden werden kann, das in einem der Verzeichnisse der Umgebungsvariable `PATH` enthalten ist.
Die Funktionen dieser Bibliothek lassen sich in 4 Gruppen unterteilen, die wir nun der Reihe nach vorstellen werden. Die meisten Funktionen geben im Erfolgsfall `OVID_OK` zurück (auf Abweichungen werden wir im folgenden hinweisen) und im Fehlerfall einen der Fehlercodes, die zu Beginn der Datei `admininterface.h` aufgelistet sind und deren Bezeichnung selbsterklärend ist.

3.1.4.1 Verbindung zur SecOVID-Datenbank

Mit folgenden Funktionen wird eine Verbindung zur SecOVID-Datenbank aufgebaut:

```
int
SecOVID_connectCard(const char *adminname, const char *server,
                    const char *port, const unsigned short CTN,
                     SecOVIDConnHandle* DEFAULT(connection, NULL));
int
SecOVID_connectSoft(const char *adminname, const char *server,
                    const char *port, const char *tokenfile, const char *PIN,
                    SecOVIDConnHandle* DEFAULT(connection, NULL));
int
SecOVID_connectData(const char *adminname, const char *server,
                    const char *port, char *softdata, const char *PIN,
                    SecOVIDConnHandle* DEFAULT(connection, NULL));
```

Diese Funktionen bauen für den Administrator mit dem angegebenen Namen eine Verbindung zur Datenbank auf dem angegebenen Server auf. Dabei kann für `server` entweder der Hostname oder die IP-Adresse übergeben werden. Für `port` muß die Nummer des Ports übergeben werden, zu dem die Verbindung hergestellt werden soll, z.b. `"1112"`.
Zur Authentifizierung kann entweder eine Chipkarte in einem Kartenleser, zu dem bereits eine Verbindung besteht (siehe unten), benutzt werden, oder wahlweise eine Datei mit einem SoftToken (in diesem Fall ist der

Dateiname zu übergeben) oder der Inhalt einer solchen Datei. In beiden Fällen ist die PIN zur Enschlüsselung des SoftTokens mit zu übergeben, während wir im Fall der Chipkarte davon ausgehen, dass die PIN der Karte bereits verifiziert wurde.
Der letzte Parameter dient im wesentlichen der Unterstützung einer möglichen zukünftigen Erweiterung der Bibliothek, zur Zeit kann immer nur eine Datenbank-Verbindung benutzt werden, in Zukunft soll es möglich sein, mehrere Verbindungen gleichzeitig zu benutzen, zum Beispiel um Benutzerdaten leicht von einem Server zu einem anderen Server zu kopieren. Zur Unterstützung dieser Möglichkeit können sie sich jeweils einen **connection handle** zurückgeben lassen, der dann in allen weiteren Funtionsaufrufen benutzt werden sollte. Wenn Sie sicher sind, dass Sie auch zukünftig mit einer Datenbankverbindung auskommen, können Sie hier stets NULL übergeben bzw. in C++ diesen letzten Parameter einfach weglassen.
Um die Verbindung zur Datenbank wieder zu schließen, benutzen Sie die Funktion

```
int
SecOVID_close(SecOVIDConnHandle DEFAULT(connection, NULL));
```

Für bestehende Verbindungen können Sie mit den Funktionen

```
int
SecOVID_getAdminName(char *admin, SecOVIDConnHandle DEFAULT(connection, NULL));
int
SecOVID_getAdminGroups(int *grp, SecOVIDConnHandle DEFAULT(connection,
NULL));
```

auf den Namen des Administrators, der die Verbindung aufgebaut hat bzw. auf seine Gruppenzugehörigkeit zugreifen. Dabei sollten die Parameter `admin` bzw. `grp` jeweils auf den Beginn eines ausreichend großen Arrays zeigen, für `admin` also typischerweise wenigstens 16 Byte und für `grp` wenigstens 8 Byte.

3.1.4.2 Verbindung zum Kartenleser

Um die schon angesprochene Verbindung zum Kartenleser aufzubauen bzw. abzubauen und im Bedarfsfall die PIN zu verifizieren bzw. mit Hilfe der PUK freizuschalten, stehen die folgenden Funktionen zur Verfügung.

```
DLLIMPEXP int
SecOVID_SearchCardTerminalIndex();
int
SecOVID_SearchAllCardTerminalIndices();
void
SecOVID_SetCardTerminalIndex(int index);
int
SecOVID_OpenCardTerminalConnection(unsigned short CTN);
int
SecOVID_CloseCardTerminalConnection(unsigned short CTN);
int
SecOVID_GetCardTerminalStatus(unsigned short CTN);
int
SecOVID_GetCardTerminalHasPinPad(unsigned short CTN);
int
SecOVID_ResetChipcard(unsigned short CTN);
int
SecOVID_VerifyCardPIN(unsigned short CTN,
                      unsigned char pin_len, const unsigned char *pin);
int
SecOVID_UnblockCardPIN(unsigned short CTN,
                       unsigned char puk_len, const unsigned char *puk,
                       unsigned char pin_len, const unsigned char *pin);
```

Dabei dienen die ersten drei Funktionen dazu, nachzusehen, an welchem Port ein eventueller Kartenleser angeschlossen ist bzw. auszuwählen, welcher Port für die weiteren Befehle benutzt werden soll, dabei findet `SecOVID_SearchCardTerminalIndex` den ersten freien Leser, legt diesen gleichzeitig als den zu verwendenden Leser fest und gibt den Index des Lesers zurück. Hierbei zeigt - abweichend von der üblichen Konvention für die Rückgabewerte - 0 an, dass kein Leser gefunden wurde.
`SecOVID_SearchAllCardTerminalIndices` gibt - ebenfalls abweichend von der üblichen Konvention - eine 32-Bit-Zahl zurück, bei der das i-te Bit (von rechts) gesetzt ist, wenn an Port i ein Leser gefunden wurde. Nach Aufruf dieser Funktion kann man mit `SecOVID_SetCardTerminalIndex` auswählen, welchen Leser man nun wirklich verwenden will.
Die nächsten beiden Funktionen öffnen bzw. schließen die Verbindung zum Kartenleser, dabei ist besonders zu beachten, dass die Verbindung immer geschlossen werden sollte, wenn der Leser vorübergehend nicht mehr benötigt wird, da andernfalls andere Applikationen nicht auf den Kartenleser zugreifen können.
`SecOVID_GetCardTerminalStatus` gibt Informationen über den Zustand des Lesers zurück, bei 0 liegt keine Karte im Leser, 3 bedeutet, dass eine Karte vorhanden ist, aber noch nicht mittels `SecOVID_ResetChipcard` aktiviert wurde, 5 bedeutet, dass eine Karte vorhanden und aktiviert ist.
`SecOVID_GetCardTerminalHasPinPad` gibt 1 zurück, wenn der Leser eine Tastatur zur PIN-Eingabe hat, 0 sonst.
Mit `SecOVID_VerifyCardPIN` bzw. `SecOVID_UnblockCardPIN` kann die PIN verifiziert bzw. freigeschaltet werden. Zu beachten ist hier insbesondere, dass die Übergabe eines NULL-Pointers für `pin` oder `puk` oder einer 0 für `pin_len` oder `puk_len` bedeuten, dass die PIN (und ggf. PUK) über die Tastatur des Lesers eingegeben werden sollen. Dies funktioniert natürlich nur dann, wenn der Leser auch eine Tastatur hat. Besondere Beachtung verdienen die Return-Werte dieser Funktionen, neben dem üblichen `OVID_OK` und einem eher allgmeinen Fehler wie `OVID_ERROR_CHIPCARD` sind auch folgende Werte möglich: `OVID_ERROR_PIN_LAST_TRY`, `OVID_ERROR_PIN_LOCKED`, `OVID_ERROR_PIN_MISSING`, `OVID_ERROR_PIN_CANCELLED` (wenn die Eingabe am Leser abgebrochen wurde), `OVID_ERROR_PIN_TIMEOUT` (wenn das Warten auf die Eingabe am Leser mit einem Timeout beendet wird) und `OVID_ERROR_PIN_MISMATCH` (wenn beim Freischalten der PIN nicht zweimal hintereinander die gleiche, neue PIN eingegeben wird).

3.1.4.3 High-Level-Operationen

Um die Funktion der Administrationstools abzubilden, stehen die folgenden Funktionen zur Verfügung:

```
int
SecOVID_getNoOfUsers(SecOVID_usertype type,
                     int *DEFAULT(groups, SecOVID_defaultGroup),
                     const SecOVIDConnHandle DEFAULT(connection, NULL));
int
SecOVID_create(SecOVID_usertype type, const char *username, char *name,
               char *vorname, const int* group, char *DEFAULT(PIN, NULL),
               int DEFAULT(pinlen, 6), char *DEFAULT(PUK, NULL),
               SecOVID_puktype DEFAULT(puk_type, SecOVID_secure),
               const SecOVIDConnHandle DEFAULT(connection, NULL));
int
SecOVID_batchcreate(SecOVID_usertype type, const char *username, int start,
                    int end, const int *group, char *DEFAULT(PIN, NULL),
                    int DEFAULT(pinlen, 6), char *DEFAULT(PUK, NULL),
                    SecOVID_puktype DEFAULT(puk_type, SecOVID_secure),
                    const SecOVIDConnHandle DEFAULT(connection, NULL));
           /* NULL for PIN/PUK implies to generate a random PIN/PUK,
              puklen = -1 implies to use the standard length implied
              by puk_type. */
int
SecOVID_LDAPcreate(const char* serverIP, int LDAPport, const char* suffix,
                   const char *filter, const char* username,
```

```
                  const char *password, const char* logfile,
                  SecOVID_LDAP_mode DEFAULT(mode, SecOVID_LDAP_ADD),
                  int DEFAULT(pinlen, 6),
                  SecOVID_puktype DEFAULT(puk_type, SecOVID_secure),
                  const SecOVIDConnHandle DEFAULT(connection, NULL));
int
SecOVID_update(SecOVID_usertype type, const char *username,
               const char *DEFAULT(newusername, NULL),
               int DEFAULT(forceNewusername, 0),
               char *DEFAULT(newname, NULL), char *DEFAULT(newvorname, NULL),
               const int *DEFAULT(newgroup, NULL), char *DEFAULT(newPIN, NULL),
               const char *DEFAULT(newtype, NULL),
               const SecOVIDConnHandle DEFAULT(connection, NULL));
          /* For each argument, NULL or -1 means: don't change */
int
SecOVID_delete (SecOVID_usertype type, const char *username,
                const SecOVIDConnHandle DEFAULT(connection, NULL));
int
SecOVID_addAlias (const char *username, const char *alias,
                  const SecOVIDConnHandle DEFAULT(connection, NULL));
int
SecOVID_lock (const char *username,
              const SecOVIDConnHandle DEFAULT(connection, NULL));
int
SecOVID_unlock (const char *username,
                const SecOVIDConnHandle DEFAULT(connection, NULL));
int
SecOVID_resync (const char *username,
                const SecOVIDConnHandle DEFAULT(connection, NULL));
int
SecOVID_deleteCard(SecOVID_usertype type, const char *username,
                   PINVerificationCallback PINCallback, unsigned short CTN,
                   const SecOVIDConnHandle DEFAULT(connection, NULL));
int
SecOVID_deleteGenerator(SecOVID_usertype type, const char *username,
                        PINVerificationCallback PINCallback,
                        unsigned short CTN, int DEFAULT(gen_nr, 0),
                        const SecOVIDConnHandle DEFAULT(connection, NULL));
                    /* gen_nr automatically forced to 9, if type == admin */
int
SecOVID_unblockPIN(SecOVID_usertype type, const char *username,
                   PINUnblockCallback PINCallback, unsigned short CTN,
                   const SecOVIDConnHandle DEFAULT(connection, NULL));
int
SecOVID_generateCard(SecOVID_usertype type, const char *username,
                     PINVerificationCallback PINCallback, unsigned short CTN,
                     int DEFAULT(gen_nr, 0), int DEFAULT(forceGen, 0),
                     SecOVID_gentype DEFAULT(gen_type, SecOVID_LENGTH_8),
                     const SecOVIDConnHandle DEFAULT(connection, NULL));
                    /* gen_nr automatically forced to 9, if type == admin */
int
SecOVID_importDB(const char *importfile, const char *logfile,
                 SecOVID_usertype type, int InsertCardCallback(int),
```

```
                unsigned short DEFAULT(CTN, -1),
                const SecOVIDConnHandle DEFAULT(connection, NULL));
int
SecOVID_exportDB(const char *exportfile, const char *logfile,
                SecOVID_usertype type, const char* const* DEFAULT(names,NULL),
                unsigned short DEFAULT(CTN, -1),
                const SecOVIDConnHandle DEFAULT(connection, NULL));
                /* Use names=NULL to export _all_ users/admins */
int
SecOVID_generateSofttoken(SecOVID_usertype type, const char *username,
                          const char *tokendir, const char *pinfile,
                          unsigned short DEFAULT(CTN, -1),
                          SecOVID_softtype DEFAULT(st, SecOVID_Softtoken_PIN),
                          int DEFAULT(add_to_file, 1),
                          int DEFAULT(weak, 0),
                          const SecOVIDConnHandle DEFAULT(connection, NULL));
int
SecOVID_generateEmergencyStatic(const char *username, char *password,
                                int DEFAULT(hours, 1),
                                const SecOVIDConnHandle DEFAULT(connection,
                                                                NULL));
```

Diese Funktionen entsprechen genau den Funktionen des Kommanozeilen-Administrationstools und sollten somit keiner detaillierteren Erklärung bedürfen. Daher beschränken wir uns darauf, auf einige Besonderheiten hinzuweisen:

- `SecOVID_getNoOfUsers` gibt, falls der Rückgabewert nicht negativ ist, die Anzahl der Benutzer bzw. Administratoren in den spezifizierten Gruppen zurück. Ein negativer Rückgabewert enspricht einem der üblichen Fehlercodes.
- `SecOVID_generateSofttoken` schreibt den Usernamen und die zugehörige PIN in die mittels `pinfile` angegebene Datei. Dabei bedeutet ein von 0 verschiedener Wert für den Parameter `weak`, dass das Soft-Token ohne Zuhilfenahme von Zufallszahlen von der Chipkarte mittels eines softwarebasierten Pseudo-Zufallszahlen-Generators generiert wird. Diese Option ist vom Standpunkt der Sicherheit nicht wirklich empfehlenswert, in der Praxis aber häufig nützlich, wenn gerade kein Kartenleser zur Hand ist.

3.1.4.4 Low-Level-Zugriff

Sollten diese Funktionen nicht ausreichen, gibt es auch die Möglichkeit, auf die einzelne Komponenten des Datensatzes zuzugreifen und diese zu modifizieren. Dazu dienen die Funktionen:

```
int
SecOVID_newUser(SecOVIDUserHandle* user, SecOVID_usertype type);
int
SecOVID_getShortUser(SecOVIDUserHandle* user, SecOVID_usertype type, int index,
                     int DEFAULT(sort_index, 0),
                     int *DEFAULT(groups, SecOVID_defaultGroup),
                     const SecOVIDConnHandle DEFAULT(connection, NULL));
int
SecOVID_getUser(SecOVIDUserHandle* user,
                const SecOVID_usertype type, const char* username,
                const SecOVIDConnHandle DEFAULT(connection, NULL));
int
SecOVID_fillUser(SecOVIDUserHandle user,
                 const SecOVIDConnHandle DEFAULT(connection, NULL));
```

```
int
SecOVID_writeUser(const SecOVIDUserHandle user,
                  const SecOVIDConnHandle DEFAULT(connection, NULL));
int
SecOVID_freeUser(SecOVIDUserHandle user);
int
SecOVID_isInGroup(SecOVIDUserHandle user, const SecOVIDConnHandle connection);
const int SecOVID_getMaxInfoSize();
int
SecOVID_getInfo(const SecOVIDUserHandle user, const SecOVID_UserComp id,
                unsigned long* len, void* data);
int
SecOVID_setInfo(SecOVIDUserHandle user, const SecOVID_UserComp id,
                unsigned long* len, const void* data);
```

Hinweis: Beachten Sie, dass serverseitig eine Plausibilitätsprüfung der Änderungen, die Sie an einem Benutzerdatensatz vorgenommen haben, stattfindet. Operationen, die mit den normalen Administrationstools nicht möglich sind (z.B. gleichzeitige Änderung des Vornamens und Freischaltung eines gesperrten Users), werden normalerweise als ungültig abgelehnt.
Die Funktion `SecOVID_newUser` erzeugt einen neuen, „leeren“ Benutzer und gibt im ersten Parameter einen Hanlde auf diesen zurück. Diese Funktion wird eigentlich nur zum Anlegen eines neuen Benutzers benötigt.
`SecOVID_getShortUser` entnimmt die wichtigsten Daten eines Benutzers oder Adminsitrators aus der Datenbank und gibt ebenfalls als ersten Parameter einen Handle auf diesen Datensatz zurück, mit `SecOVID_fillUser` können diese Daten bei Bedarf (etwa bevor eine Änderung vorgenommen und die Änderung zurückgeschrieben wird) vervollständigt werden. Dabei geben die Parameter an, der wievielte Benutzer oder Administrator aus den übergebenen Gruppen gemeint ist, wobei man mit dem Parameter `sort_index` spezifizieren kann, nach welchem Kriterium die Benutzer bzw. Administratoren durchnummeriert werden sollen. Dabei stehen die Zahlen 1 bis 6 für die Sortierkriterien Benutzername, Nachname, Vorname, Gruppenzugehörigkeit, Tokentyp und Status. Normalerweise erfolgt die Sortierung in aufsteigender Reihenfolge, bei Verwendung von -1 bis -6 wird in absteigender Reihenfolge sortiert.
`SecOVID_getUser` liefert gleich den vollständigen Datensatz, benötigt aber als Parameter den Benutzer- bzw. Administratornamen.
Mit `SecOVID_writeUser` wird der evtl. modifizierte Datensatz wieder in die Datenbank zurückgeschrieben.
`SecOVID_freeUser` gibt den Speicherplatz für einen Benutzerdatensatz wieder frei - jedem Aufruf von `SecOVID_newUser`, `SecOVID_getShortUser` oder `SecOVID_getUser` sollte daher ein Aufruf von `SecOVID_freeUser` entsprechen.
Mit `SecOVID_isInGroup` kann z.B. überprüft werden, ob die Rechte, die der Administrator der bestehenden Verbindung zum Datenbankserver hat, ausreichen, um den Benutzer oder Administrator wieder in die Datenbank geschrieben werden kann. Der Rückgabewert dieser Funktion ist 1, wenn die Rechte ausreichen, und 0 sonst.
`SecOVID_getMaxInfoSize` gibt zurück, wieviel Speicher ein Rückgabewert von `SecOVID_getInfo` höchstens benötigt.
Mit `SecOVID_getInfo` kann jede Komponente eines Datensatzes abgefragt werden, dabei ist zu beachten, dass `data` auf einen hinreichend großen Speicherbereich zeigen muss, dessen initiale Größe im Parameter `len` mitgegeben wird. Nach dem Aufruf ist der Wert von `len` so geändert, dass er nun anzeigt, wieviel Bytes die Antwort tatsächlich belegt. Besonders beachtenswert sind die möglichen Rückgabewerte `OVID_ERROR_INCOMPLETE_DATA`, der anzeigt, dass zunächst noch `SecOVID_fillUser` aufgerufen werden muss, um den Datensatz zu vervollständigen, bevor der Inhalt der gewünschten Komponente abgefragt werden kann, sowie `OVID_USER_ERROR_BUFFER_TOO_SMALL`, der anzeigt, dass in `data` nicht genug Speicher zur Verfügung steht, um die gewünschte Komponente zu speichern - in diesem Fall enhält `data` soviele Bytes der Komponente, wie hineinpassen und `len` wird auf den Wert gesetzt, der benötigt wird um die Komponente komplett speichern zu können. Insbesondere kann diese Funktion mit dem Wert `NULL` für `data` aufgerufen werden, woraufhin nur der benötigte Speicherplatz ermittelt und in `len` zurückgegeben wird.
Mit `SecOVID_setInfo` kann in analoger Weise der Inhalt einer Komponente überschrieben werden, dabei wird im Parameter `len` übergeben, wieviele Bytes gespeichert werden sollten und nach dem Auruf enthält die Variable

die Anzahl der tatsächlich geschriebenen Bytes. Für den Aufruf mit `data = NULL` gilt die gleiche Konvention wie bei der vorangegangenen Funktion.

3.1.5 Konfigurationsbeispiele für FreeRADIUS um weitere Möglichkeiten von RADIUS zu unterstützen

Hier wollen wir einige Beispiele zeigen, wie FreeRADIUS konfiguriert werden kann, um mit SecOVID zusammenzuarbeiten; dazu wollen wir auf drei Teilbereiche eingehen, nämlich Accounting, EAP und die Verwendung des Users-Files (bzw. der User-Datenbank) von FreeRADIUS. Die Idee ist jeweils, FreeRADIUS seine Arbeit tun zu lassen und die eigentliche Authentifikation als Proxy-Request an den SecOVID-Server weiterzuleiten. Im wesentlichen ist diese Möglichkeit schon in FreeRADIUS angelegt, problematisch ist in dieser Hinsiche eigentlich nur das Sub-Protokoll EAP-MD5 aus der EAP-Protokollfamilie. Deshalb haben wir hier ein spezielles Modul entwickelt (vgl. Abschnitt 2.1.4), das analog zu anderen EAP-Sub-Protokollen die eigentliche Authentifikationsanfrage aus der „Verpackung“ in Form des EAP-Protokolls auspackt und an einen anderen Server weiterleiten kann. Das bei FreeRADIUS derzeit mitgelieferte EAP-MD5-Modul nimmt die Passwortverifikation immer lokal vor.
Um FreeRADIUS und SecOVID auf demselben Rechner nutzen zu können, beachten Sie bitte folgende Hinweise zur Konfiguration:

1. Im Normalfall wollen Sie FreeRADIUS als von außen sichtbaren Server konfigurieren, dieser sollte also die Standard-Ports 1812 (für `radius`) bzw. 1813 (für `radacct`) benutzen. Tragen Sie dazu in

   `/usr/local/etc/raddb/radiusd.conf`

   Port 1812 ein, d.h. ersetzen Sie die vorhanden Zeile `port = 0` durch `port = 1812`.

2. Wählen Sie den internen, d.h. den für die Weiterleitung von FreeRADIUS zu SecOVID, benutzten Port, z.B. 1645. Da SecOVID immer den in der Datei `/etc/services` eingetragenen Port benutzt, müssen Sie den dortigen Eintrag für `radius` also auf 1645 ändern.

3. Legen Sie ein Geheimnis für die Kommunikation zwischen FreeRADIUS und SecOVID fest. Tragen Sie das Geheimnis einerseits in der Datei `/etc/SecOvid/clients` ein, z.B.

   ```
   localhost  geheim
   ```

4. Konfigurieren Sie die Weiterleitung für FreeRADIUS, indem Sie die Datei `/usr/local/etc/raddb/proxy.conf` bearbeiten. Sie können z.B. am Ende folgendes anhängen:

   ```
   realm SecOVID {
         type        = radius
         authhost    = localhost:1645
         secret      = geheim
   }
   ```

Nun können Sie den SecOVID-Server und den FreeRADIUS-Server starten. Zum Testen sollten Sie nun Port und Geheimnis von FreeRADIUS verwenden. Die Geheimnisse für FreeRADIUS werden in

`/usr/local/etc/raddb/clients.conf`

konfiguriert.
Um nun z.B. nur Accounting mit FreeRADIUS zu behandeln und alle Authentifikationsanfragen an den SecOVID-Server weiterzuleiten, tragen Sie in der Datei `/usr/local/etc/raddb/users` folgendes zu Anfang ein:

```
DEFAULT    Packet-Type==Access-Request, Proxy-To-Realm := SecOVID
```

Um z.B. EAP mit FreeRADIUS zu behandeln, und die eigentliche Authentifizierung mit SecOVID vorzunehmen, lassen wir den FreeRADIUS-Server alle Requests behandeln, die schon als EAP-Anfragen erkannt wurden oder das Attribute `EAP-Message` enthalten:

```
DEFAULT Auth-Type != EAP, EAP-Message !* ANY, Proxy-To-Realm := SecOVID
```

Zusätzlich sollten Sie, wenn Sie PEAP benutzen wollen, den Eintrag

```
proxy_tunneled_request_as_eap = no
```

in der Datei `/usr/local/etc/raddb/eap.conf` innerhalb des `peap{}` Abschnitts eintragen. Beachten Sie auch, das für EAP-TTLS und PEAP einige Konfigurationsarbeit am FreeRADIUS-Server erforderlich ist, insbesondere muss ein Server-Zertifikat mit zugehörigen Schlüsseln erzeugt werden, und diese Dateien müssen in den entsprechenden Abschnitten der Datei `/usr/local/etc/raddb/eap.conf` eingetragen werden.

3.1.6 Erstellung der SoftToken-Applikationen für Mobiltelefone und PDAs

Den SecOVID-Midlet-Generator verwenden Sie, um aus erstellten SoftToken-Datensätzen (*.kst), welche mittels SecOVID-Admintool exportiert wurden, SoftToken-Applikationen, im folgenden „Midlet“ genannt, zu erstellen. Diese Midlets können dann auf allen Geräten ausgeführt werden, auf denen ein Runtime-Environment der J2ME (Java 2 Micro Edition) installiert ist. In diesem Fall handelt es sich dabei um java-fähige Mobiltelefone und PDAs mit dem Betriebssystem PalmOS ab Version 3.5.
Bei PDAs mit den Betriebssystemen Windows CE bzw. Pocket PC, ist dieser Zwischenschritt nicht notwendig und Sie können direkt zu Abschnitt 3.1.7.4 gehen.
Zur Erstellung der Midlets stehen Ihnen zwei Aufrufmöglichkeiten zur Auswahl. Sie können entweder genau ein Midlet für genau einen Benutzer erzeugen, oder Sie können mehrere Midlets für mehrere Benutzer gleichzeitig erzeugen. Dazu sammeln Sie die zuvor erstellten SoftToken-Datensätze (*.kst) in einem Verzeichnis. Der Vorgang zur Erstellung wird anschließend mittels eines ANT-Skriptes, welches im Verzeichnis des MidletGenerators zu finden ist, angestoßen. Die dazu erforderliche Umgebung wurde von Ihnen, wie in Abschnitt 2.1.5 beschrieben, eingerichtet.
Folgende Aufrufmöglichkeiten stehen Ihnen nun zur Verfügung:

1. UNIX
   - Erstellen der Applikation anhand einer einzelnen Tokendatei:
     - Pfade zur Installation in der Datei buildconfig.properties anpassen
     - `ant buildFILE -DKSTFILE="<Verzeichnis>/*.kst"`
   - Erstellen mehrerer Applikationen anhand eines Verzeichnisses mit Tokendateien.
     - Pfade zur Installation in der Datei buildconfig.properties anpassen
     - `ant buildDIR -DKSTDIR="<Verzeichnis>"`
2. WIN32
   - Erstellen der Applikation anhand einer einzelnen Tokendatei:
     - Pfade zur Installation in der Datei buildconfig.properties anpassen
     - `ant buildFILE -DKSTFILE="<Verzeichnis>\*.kst"`
   - Erstellen mehrerer Applikationen anhand eines Verzeichnisses mit Tokendateien.
     - Pfade zur Installation in der Datei buildconfig.properties anpassen
     - `ant buildDIR -DKSTDIR="<Verzeichnis>"`

Bei diesem Vorgang wird jeweils automatisch eine neue Verzeichnisstruktur erstellt, in der für jeden Benutzer ein gleichnamiges Verzeichnis angelegt wird, in dem das Midlet für Handys und das Midlet für PalmOS-Geräte abgelegt ist. Diese Verzeichnisse finden Sie auf UNIX-Systemen in dem Ordner
`"<IhrPfad>/MidletGenerator/Release"`, bzw. auf Windows Systemen unter
`"<IhrPfad>\MidletGenerator\Release"`.

3.1.7 Distribution der SoftToken-Applikation zu Mobiltelefonen und PDAs

Bei Anwendungen dieser Art muss die Sicherheit gewährleistet sein. Es dürfen keine Daten unbeabsichtigt nach außen gelangen. Dies zählt zum einen für die Anwendung selbst und zum anderen für die Übertragung der Applikation zum Endgerät (Mobiltelefon oder PDA). Diese Übertragung des Midlets zum Endgerät soll nachfolgend geschildert werden. Es werden ausschließlich sichere Übertragungsarten verwendet. Bei dem WAP-Download auf Handys nennt sich diese Ubertragungsart WTLS, welche ein Äquivalent zu SSL darstellt. Bei den PDAs werden die Midlets durch eine Synchronisation mit dem PC auf das Endgerät transferiert und gelangen somit nicht in öffentliche Netze. Der Download der Applikation zum PC wird über SSL abgesichert.

3.1.7.1 Vorbereitung für die Distribution

Um diese Techniken einsetzen zu können, muss natürlich ein dementsprechend konfigurierter Server verwendet werden. In diesem Fall benutzen wir den Apache-Tomcat-Server, welcher zum einen als ganz normaler WEB-Server eingesetzt werden kann, zudem aber auch die Fähigkeit besitzt, mit Servlets umzugehen, welche wir zur Zugangskontrolle und Weiterleitung der Benutzer verwenden. Haben Sie die Midlets nach der in Abschnitt 3.1.6 beschriebenen Vorgehensweise erstellt, müssen die Verzeichnisse und die Datei access.dat auf dem Tomcat-Server abgelegt werden. Die Datei access.dat wurde beim Erstellen der SecOVID-SoftToken-Applikationen automatisch erzeugt und enthält die Benutzernamen und die zugehörigen Passworte. Sie wird im Verzeichnis TOMCAT_HOME/webapps/examples/WEB-INF/classes/Kobil abgelegt. Dort befindet sich ebenfalls eine Datei namens *user.dat*, in der automatisch vom Servlet jeder User eingetragen wird, der sein Midlet bereits heruntergeladen hat. Die Benutzerverzeichnisse mit den Midlets werden in TOMCAT_HOME/webapps/ROOT/release abgelegt.

3.1.7.2 Download per Mobiltelefon

Der Benutzer eines Handys muss nun eine WAP-Verbindung zum Server aufbauen, indem er die passende URL eingibt (Bsp.: https://wap.kobil.de/index.wml). Die Einstellungen, um eine WAP-Verbindung aufzubauen, werden durch den Handy-Provider vorgegeben. Normalerweise sind Handys bereits durch den Provider vorkonfiguriert. Eventuell muss man per Hand noch die Verbindungssicherheit aktivieren. Steht die Verbindung zum Server, so wird der Benutzer durch die Startseite aufgefordert, seine Benutzerdaten einzugeben. Diese Daten wurden ihm durch einen PIN-Brief, welcher vom Admintool erstellt wurde, mitgeteilt. Gibt er die Zugangsdaten korrekt ein, wird er vom Servlet zu seinem Verzeichnis weitergeleitet, von wo er dann das Midlet beziehen kann. Nach dem Download befindet sich (abhängig vom jeweiligen Mobiltelefon) unter den Applikationen des Mobiltelefons eine weitere Anwendung namens **Secovid**. Führt man diese Anwendung aus, erscheint ein Menü, welches die Einträge „Pin ändern“ und „OTP erzeugen“ enthält. Wählt man „Pin ändern“, muss erst die alte und dann zweimal die neue Pin eingegeben werden. Es erfolgt eine Meldung, die die Änderung bestätigt. Möchte man ein neues OTP generieren, muss man die aktuelle PIN eingeben worauf das nächste gültige OTP angezeigt wird.

3.1.7.3 Download per PDA unter PalmOS

Benutzer von Palm-Geräten laden sich ihre Applikation vom gleichen Server herunter, indem sie eine Verbindung mittels eines normalen Browsers herstellen. Der Server erkennt, dass der Verbindungswunsch von keinem Handy kommt und leitet den Benutzer dementsprechend zu dem Verzeichnis weiter, in dem die Applikation für Palm-PDAs liegt. Hat der Benutzer die Anwendung heruntergeladen, kann er sie über die gewohnte Synchronisationsmethode auf dem PDA installieren.

Installationsvoraussetzung:
Es muss eine Java-Runtime auf dem Gerät installiert werden um die Applikation benutzen zu können.

Für die Palm OS Version 3.5 beziehen Sie bitte das Paket „MIDP for Palm OS“:
http://java.sun.com/products/midp4palm/
Für die Palm OS Version 5.2 beziehen Sie bitte das Paket „IBM's WebSphere Micro Environment“:

http://www.palmone.com/us/support/jvm/

3.1.7.4 Download per PDA mit den Betriebssystemen Windows CE / PocketPC

Im Gegensatz zu PDAs die unter dem Betriebssystem Palm OS laufen, benötigen wir für PDAs mit den Betriebssystemen Windows CE bzw. Pocket PC keine spezielle Applikation. Für diese Geräte gibt es eine Applikation die für alle Benutzer identisch ist. Diese Applikation ist, zusammen mit einer kurzen Bedienungsanleitung im winCE-Verzeichnis der SecOVID-CD enthalten. Natürlich muss man auch hier eine Personalisierung durchführen. Dies geschieht, indem man den zuvor exportierten geheimen SecOVID-SoftToken-Datensatz auf das zu personalisierende Gerät kopiert. Während der Installation der Applikation auf Ihrem PDA wird in dem nichtflüchtigen Speicherbereich ein Verzeichnis namens **Secovid** angelegt. Nichtflüchtiger Speicherbereich bedeutet, dass es sich dabei um einen Speicherbereich handelt, der auch ohne Stromversorgung die Daten sicher behält. Genau in dieses Verzeichnis kopieren Sie den geheimen SoftToken-Datensatz. Ab diesem Moment ist der PDA einsatzbereit und kann unter Angabe der Benutzer-PIN OTPs generieren und anzeigen.

3.1.7.5 Installation des Tomcat-Servers

Auf Ihrer Installations-CD finden Sie die Datei `$UNIX/Softtoken/install\_Tomcat.sh`. Beim Aufruf dieses Skriptes wird zunächst das J2SDK1.4 installiert, welches sich dann im Verzeichnis /usr/java befindet. Danach wird jakarta-tomcat-4.1.24 nach /usr/local kopiert. In diesem Verzeichnis gibt es eine Datei jakarta-tomcat-4.1.24/bin/catalina.sh, in der die JAVA_HOME Variable gesetzt ist. Wurde das J2SDK NICHT nach /usr/java installiert, ist dieser Eintrag dort zu ändern. Mit dem Befehl: /usr/local/jakarta-tomcat-4.1.24/bin/startup.sh kann der Server gestartet werden. Dabei ist darauf zu achten dass der SSL Port 443 von keiner anderen Anwendung bereits verwendet wird. Wenn Sie einen Testbetrieb im ungesicherten Modus durchlaufen möchten, muss zusätzlich der http-Standard-Port 80 frei sein. Sofern Sie noch keinen Keystore erstellt haben, wird der Server nicht ordnungsgemäß laufen. Um diesen zu erstellen, können Sie auf das im Lieferumfang befindliche Skript *keystore.sh* zurückgreifen, welches mittels „keytool“ einen neuen Keystore erstellt, den erforderlichen Server Request erzeugt und später den Keystore für den Einsatz im Tomcat-Server konfiguriert. Haben Sie den Server Request von einer anerkannten CA zertifizieren lassen, bekommen Sie ein neues Zertifikat, welches als RootCA eine der von Ihrem verwendetem WAP-Gateway akzeptierten Zertifikate besitzen muss. Im Zusammenhang mit diesem neuen Zertifikat erhalten Sie meist auch die Zwischenzertifikate, die Sie benötigen, um dem Gateway die Vervollständigung der Zertifikatskette zu ermöglichen. Um diese Kette nun im Keystore zu installieren, benutzen Sie bitte das o.g. Skript *keystore.sh*. Wurde der Keystore erfolgreich eingerichtet, kann dieser im TOMCAT_HOME-Verzeichnis abgelegt werden. In der Datei /usr/local/jakarta-tomcat-4.1.24/conf/server.xml müssen Sie die Pfadangabe des Keystores eventuell anpassen.
Weitere Informationen entnehmen Sie bitte der Dokumentation des Tomcat-Servers.

3.1.7.6 Installation von Zertifikatsketten im Keystore des Tomcat-Servers

Ein Keystore enthält verschiedene Einträge, die sich in die Hauptgruppen *keyEntry* und *trustcacerts* aufteilen. Die letztere enthält Zertifikate, die als vertrauenswürdig eingestuft werden und dienen der Client Authentifikation, auf die wir aber verzichten können. Der *keyEntry* hingegen ist für die Speicherung unseres Serverzertifikats in Verbindung mit den Schlüsseln zuständig. Dieser Inhalt wird beim Aufbau einer SSL-Verbindung zwischen den Kommunikationspartnern ausgetauscht und ermöglicht somit den sicheren Datenaustausch. Da in unserem Fall dieser Verbindungsaufbau zwischen einem WAP-Gateway und dem Tomcat Server stattfindet, müssen bestimmte Anforderungen des WAP-Gateways erfüllt werden. Das WAP-Gateway von T-Mobile stellt dabei folgende Anforderungen, um eine Verbindung aufzubauen:

1. Anfragen an den Server werden standardmäßig an Port 80 bzw. 8080 für eine ungesicherte HTTP-Verbindung gestellt.
2. Anfragen bei gesicherten HTTPS-Verbindungen werden standardmäßig an Port 443 gestellt.
3. Bei gesicherten HTTPS-Verbindungen muss das austellende Root-Zertifikat vom Gateway als vertrauenswürdig eingestuft sein.

4. Werden auf dem Server access-Listen verwendet, so müssen bestimmte IP-Adressen über die das Gateway kommuniziert, Zugriff erhalten.

(Genauere Informationen erhalten Sie vom Betreiber des Gateways.)

Insbesondere Punkt 3 muss besondere Beachtung finden, da es in manchen Fällen nicht ausreicht, das passende Root-Zertifikat zu verwenden. Ein solcher Fall liegt dann vor, wenn das Serverzertifikat nicht direkt von einem der Root-Zertifikate signiert wurde, sondern eine Zwischeninstanz enthält. Es kann sein, dass dieses Zertifikat der Zwischeninstanz im Gateway unbekannt ist und somit der Verbindungswunsch seitens des Gateways abgelehnt wird. Um ein Zwischenzertifikat im Gateway bekannt zu machen, muss der Tomcat Server dieses beim Verbindungsaufbau übertragen. Somit kann das Gateway die Zertifikatskette bis zur Root vervollständigen, und die Verbindung wird akzeptiert. Das Übertragen solcher Zertifikatsketten wird im Keystore ermöglicht, indem der keyEntry nicht nur das Serverzertifikat, sondern eben eine solche Zertifikatskette enthält. Da es mit dem keytool allerdings nicht möglich ist, eine solche Zertifikatskette zu importieren, muss man das Skript *keystore.sh* verwenden, um den Keystore anzupassen. Beim Erzeugen des Server Requests wurde bereits ein Keystore erstellt, der ein dummy-Zertifikat und den privaten Schlüssel enthält. Nachdem der Request von einer Zertifizierungsstelle signiert ist, erhalten Sie als Antwort ein Serverzertifikat und die benötigten Zwischenzertifikate bis zum Root-Zertifikat. Diese Zertifikate liegen in einer Datei vor, die z.B. folgendes Aussehen hat:

userca:
—BEGIN CERTIFICATE—
...
...
—END CERTIFICATE—
serverca:
—BEGIN CERTIFICATE—
...
...
—END CERTIFICATE—
rootca:
—BEGIN CERTIFICATE—
...
...
—END CERTIFICATE—

Das Skript *keystore.sh* kann aus einem gegebenen Keystore in Verbindung mit dieser Zertifikatsdatei einen neuen Keystore erstellen, welcher im Tomcat-Server verwendet werden kann.

Benutzung des Skriptes keystore.sh
Das keystore - Skript kann mit drei verschiedenen Parametern aufgerufen werden. Mit dem Parameter *-help* wird eine kurze Anleitung ausgegeben. Wird keystore.sh mit dem Parameter *-request* aufgerufen, so wird ein neues Schlüsselpaar generiert, es wird ein neuer Keystore angelegt, und es wird eine Datei namens *request.pem* erstellt, die zur Zertifikatsbeantragung benötigt wird. Der letzte und eigentlich wichtigste Parameter ist *-certchain*. Dieser Parameter wird dann benötigt, wenn Sie von Ihrer Zertifizierungsstelle ein Serverzertifikat und die benötigten Zwischenzertifikate erhalten. Um diese Zertifikatskette nun in den Keystore zu importieren, müssen Sie drei weitere Parameter angeben, die den Pfad zu dieser Zertifikatsdatei, die Anzahl der darin befindlichen Zertifikate und das Passwort des ursprünglichen Keystores enthalten. Beispiel:

```
keystore.sh -certchain ./DownloadCert.pl 3 changeit
```

Wurde der Keystore erstellt und abgespeichert, muss man in der Datei TOMCAT_HOME/conf/server.xml die Passage in der Port 443 aktiviert wird (Connector für SSL) einkommentieren und den Pfad zur Keystore-Datei anpassen.

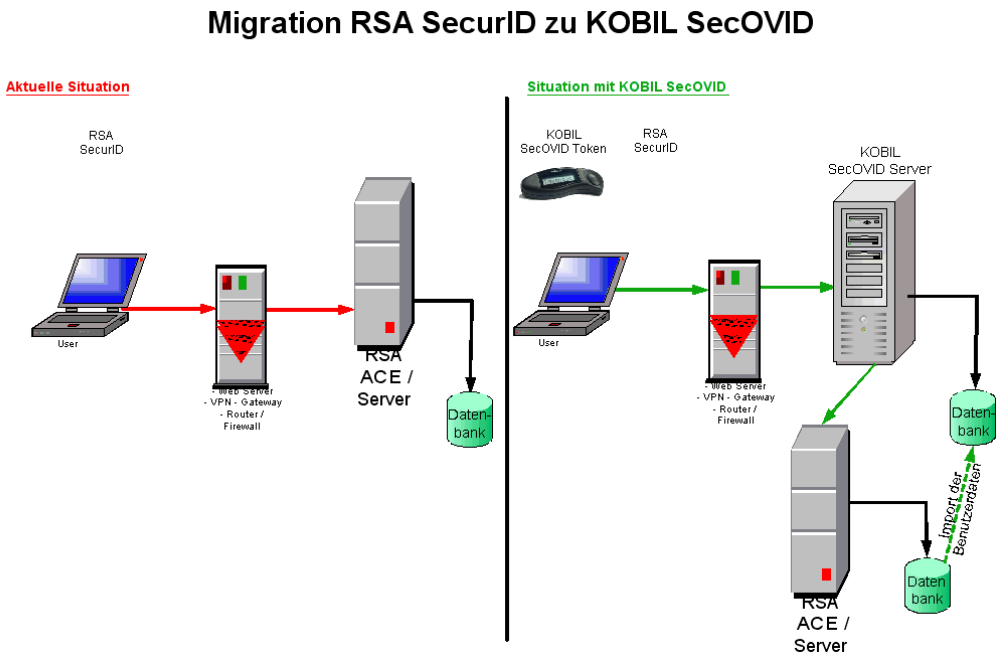
3.1.8 Migration von Fremdauthentifikationssystemen auf SecOVID

KOBIL SecOVID bietet die Möglichkeit, Benutzer des Authentifikationssystems RSA Security ACE/ Server (z.B. securID Tokens) mit minimalem administrativem Aufwand auf das SecOVID-System zu migrieren.[1]
Hierzu wird der KOBIL SecOVID-Server als Proxy-Server zwischen alle Clients des RSA ACE/ Servers (z.B. Router, Firewall, VPN-Gateway oder Webserver) und das RADIUS-Interface des ACE/ Servers geschaltet. Dazu wird die IP-Adresse des ACE/ Servers in die neu anzulegende Datei `ext_authen` eingetragen, der Inhalte sollte z.B. so aussehen:

```
ace 192.168.1.137
```

Auf Seiten des ACE/ Servers muss der SecOVID-Server als Client eingetragen werden, zu dem das Geheimnis gehört, das für den SecOVID-Server in der `clients`-Datei hinter `localhost` eingetragen ist. Außerdem ist zu beachten, dass der ACE/ Server so konfiguriert sein muss, dass er RADIUS-Anfragen an Port 1812 entgegen nimmt.
Alle ACE-Benutzer werden in die SecOVID-Datenbank aufgenommen und erhalten vorab ein SecOVID Token (oder eine SecOVID-Smart Card mit SecOVID Reader), verwenden jedoch ihr securID-Token, bis dieses ungültig wird (spätestens nach 3 Jahren). Der Benutzer bewahrt das SecOVID Token in der Zwischenzeit an einem sicheren Ort auf. Solange das SecOVID Token nicht benutzt wird, wird auch keine Batterieleistung verbraucht.
Der SecOVID-Server verarbeitet eine Passwortanfrage wie folgt: Auf Grund eines spezifischen Eintrages für den betreffenden Benutzernamen (auf dem SecOVID-System) nimmt er zunächst an, dass es sich um ein für den ACE/ Server bestimmtes Einmalpasswort bzw. Passcode handelt. Er leitet die Passwortanfrage daher zur Überprüfung an den ACE/ Server (dessen IP-Adresse er der Datei `ext_authen` entnimmt) weiter und nimmt dessen Ergebnis entgegen. Im Falle eines positiven Ergebnisses gibt der SecOVID-Server die positive Antwort an den entsprechenden Client (z.B. Firewall) zurück. Im Falle, dass der ACE/ Server den Passcode ablehnt, prüft der SecOVID-Server, ob es sich um ein korrektes SecOVID-Einmalpasswort handelt. Falls nein, gibt er eine Ablehnung an den Client zurück. Falls ja, gibt er eine entsprechende positive Meldung an den Client und setzt den Status des entsprechenden Benutzers in der SecOVID-Datenbank auf „SecOVID-Benutzer“ (vorher lautete der Status „ACE-Benutzer“). Künftig akzeptiert der SecOVID-Server für diesen Benutzer nur noch SecOVID-Einmalpasswörter; der Benutzer wurde ohne weiteren administrativen Eingriff automatisch auf das SecOVID-System migriert. Hierzu war (nach einmaligem Vorbereitungsaufwand für alle Benutzer) nichts weiter notwendig, als dass der Benutzer das ihm bereits u.U. seit langem vorliegende SecOVID Token (oder SecOVID Smart Card) benutzt.

[1] Analog kann die Migration von anderen (RADIUS-basierten) Authentifikationssystemen auf das SecOVID-System erfolgen.



Wir schildern nun detailliert, welche Schritte durch den Administrator in welcher Reihenfolge durchzuführen sind, um mit minimalem Aufwand die Migration von beispielsweise 10000 ACE-Benutzern auf das SecOVID-System vorzubereiten:

1. Importiere mit Hilfe des SecOVID-Administrationstools die geheimen SecOVID Token-Daten von 10000 SecOVID Tokens in die SecOVID-Datenbank.
2. Markiere die entsprechenden Einträge in der SecOVID-Datenbank als freie Tokendaten (Änderung des gentype mit Hilfe des SecOVID-Administrationstools (GUI)).
3. Exportiere alle ACE-Benutzer aus der Datenbank des ACE/ Servers (ASCII-Export).
4. Konvertiere die exportierten Daten mit dem dafür bestimmten KOBIL-Tool in die SecOVID-kompatible ASCII-Syntax.
5. Verknüpfe die für die 10000 Benutzer vorgesehenen freien SecOVID Token-Daten mit den 10000 Benutzereinträgen im für das SecOVID-System aufbereiteten ASCII-File. Diese Verknüpfung kann mit wenigen Mausklicks mit dem SecOVID-Administrationstool vorgenommen werden (Menüpunkt „Datenbank“, „Match to ACE“).
6. Die Konfigurationsdateien (IP-Adressen, shared secrets) werden auf allen Authentifikationsclients (Firewalls, VPN-Gateways etc.) und auf dem ACE/ Server so angepasst, dass die Authentifikationsclients ihre Passwortanfragen an den SecOVID-Server senden, und dieser als RADIUS-Authentifikationsclient für den ACE/ Server auftreten kann. Ggf. wird die RADIUS-Schnittstelle des ACE/ Servers aktiv geschaltet.

Falls erwünscht, kann die Migration der RSA securID-Benutzer auf das KOBIL SecOVID-System erfolgen, ohne einen Eingriff am laufenden RSA ACE/Server vorzunehmen. Um den SecOVID-Server netzwerktechnisch zwischen die Authentifikationsclients und den ACE/Server zu plazieren, kann die IP-Adresse des SecOVID-Servers auf allen Authentifikationsclients eingetragen werden, so dass der ACE/Server seine IP-Adresse beibehalten kann.
Auch wenn Sie vermeiden wollen, die ggf. inaktive RADIUS-Schnittstelle des ACE/Servers zu aktivieren, besteht die Möglichkeit, die Benutzer auf das SecOVID-System zu migrieren. Kontaktieren Sie in diesem Fall bitte Ihren KOBIL-Partner, der Ihnen eine geeignete Lösung vorschlagen wird.

Auf ähnliche Art und Weise kann die Migration von anderen RADIUS-basierten Authentifikationssystemen auf KOBIL SecOVID erfolgen. Kontaktieren Sie bitte Ihren KOBIL-Partner.

3.2 Clientseite

3.2.1 Konfiguration eines RADIUS-Servers*

Wie Sie Ihren RADIUS-Server (gemeint ist hier der RADIUS-Server eines anderen Herstellers) geeignet konfigurieren und im laufenden Betrieb bedienen, entnehmen Sie bitte der Dokumentation Ihres RADIUS-Servers. Der SecOVID-Server stellt bzgl. der Authentifikation und Autorisierung von Benutzern einen vollwertigen RADIUS-Server dar, der insbesondere auch statische Passwörter authentifizieren kann. Daher kann der SecOVID-Server in der Regel die heutige Rolle Ihres ggf. vorhandenen RADIUS-Servers (eines anderen Herstellers) übernehmen. Dennoch können Sie natürlich Passwortanfragen zunächst durch einen anderen RADIUS-Server entgegennehmen, um diese dann von dort automatisch an den SecOVID-Server weiterzuleiten. Wie Sie Ihren RADIUS-Server konfigurieren, um ihn als entsprechenden *Proxy-Server* zu konfigurieren, entnehmen Sie bitte dem Handbuch Ihres RADIUS-Servers. Stellen Sie sicher, dass die RADIUS-konforme Kommunikation zwischen RADIUS-Server und SecOVID-Server nicht durch eine Firewall behindert wird. Das bedeutet, die Firewall muss alle UDP-Pakete durchlassen, die vom RADIUS-Server zum SecOVID-Server auf dessen RADIUS-Port (in der Regel Port 1812, siehe den Eintrag `radius` in der Datei `/etc/services` des SecOVID-Servers) gesendet werden. Zudem muss die Firewall alle UDP-Pakete durchlassen, die vom SecOVID-Server an den RADIUS-Server (undefinierter Port) zurückgesendet werden.
Beispiel:
Nehmen wir an, ihre Endbenutzer müssten sich bei Ihrer RADIUS-fähigen Firewall authentifizieren, ehe Sie Zugang zu einem lokalen Netzwerk erhalten. Nehmen wir weiter an, ein Teil Ihrer Endbenutzer solle sich mit SecOVID-Einmalpasswörtern, ein anderer Teil solle sich mit statischen Passwörtern authentifizieren. Die Firewall soll alle Passwortanfragen an Ihren RADIUS-Server (eines anderen Herstellers als KOBIL) senden. Dieser RADIUS-Server soll dann über die Gültigkeit der statischen Passwörter selbst entscheiden, während er die Einmalpasswortanfragen an den SecOVID-Server weiterleitet.
Dann haben Sie folgendes zu tun:

- Konfiguration Ihrer Firewall
  (siehe Handbuch oder Dokumentation Ihrer Firewall):
  - Konfigurieren Sie Ihre Firewall so, dass Sie Zugriffsversuche auf das lokale Netzwerk erst nach Passwortüberprüfung zulässt.
  - Stellen Sie für alle Endbenutzer, die Zugang zum lokalen Netzwerk erhalten sollen, als Passwortmethode „RADIUS“ ein.
  - Tragen Sie als RADIUS-Server in Ihrer Firewall die IP-Adresse Ihres RADIUS-Servers ein.
  - Tragen Sie ein shared secret für den RADIUS-Server ein.
- Konfiguration Ihres RADIUS-Servers
  (siehe Handbuch zu Ihrem RADIUS-Server):
  - Tragen Sie die IP-Adresse der Firewall als RADIUS-Client ein.
  - Tragen Sie das shared secret für die Firewall ein.
  - Tragen Sie Ihre Endbenutzer in `/etc/raddb/users` ein. Definieren Sie hier insbesondere die für den jeweiligen Benutzer anzuwendende Passwortmethode bzw. im Falle, dass ein anderer Rechner (SecOVID-Server) über die Gültigkeit des Passwortes entscheiden soll, die IP-Adresse dieses Rechners (SecOVID-Server) sowie ein shared secret für diesen Rechner. (Es ist normalerweise nicht notwendig, diesen Rechner (SecOVID-Server) mit den gleichen Parametern in der Datei `/etc/raddb/clients` zu registrieren.) Im Falle des SecOVID-Servers könnte die Datei `/etc/raddb/users` wie unten angegeben aussehen, andere RADIUS-Server erwarten z.B. einen Verweis auf einen in einer separaten Konfigurationsdatei oder in einem separaten Abschnitt definierten Proxy – hier müssen Sie die Dokumentation Ihres RADIUS-Servers beachten.
  - Starten Sie den RADIUS-Server, z.B. mit dem Kommando `/etc/raddb/bin/radiusd`

  Die Datei `/etc/raddb/users` könnte beispielsweise auf einem SecOVID-Server folgende Einträge besitzen:

```
testuser1    Password = "sthX4u9K"
       Service-Type = Login-User,
       Login-Service = Telnet,
       Login-Host = 192.168.88.145

DEFAULT    Password = "PROXY"
       DTC-Auth-Server=192.168.88.164,
       DTC-Auth-Secret="wreJG67y.!tezw3"
```

Angenommen, dem RADIUS-Server würde eine `telnet`-Passwortanfrage mit dem Benutzernamen `testuser1` vorgelegt. Dann würde geprüft werden, ob das vorgelegte Passwort mit der hier gespeicherten Zeichenkette `sthX4u9K` übereinstimmt. Wird hingegen eine Passwortanfrage mit dem Benutzernamen `testuser2` an den RADIUS-Server gerichtet und es existiert kein expliziter Eintrag für `testuser2`, so kommt der DEFAULT-Eintrag zur Anwendung. Dies bedeutet, dass die vorgelegte Anfrage an den Authentifikationsserver mit der Adresse `192.168.88.164` weitergeleitet wird. Dabei werden die zu übertragenden Datenpakete gemäß RADIUS-Protokoll mit dem shared secret `wreJG67y.!tezw3` verschlüsselt. Die angegebene Adresse könnte z.B. die des SecOVID-Servers sein. Der SecOVID-Server würde die Anfrage entschlüsseln, das mitgelieferte Passwort als Einmalpasswort interpretieren und dessen Gültigkeit überprüfen. Anschließend würde er das Ergebnis der Überprüfung verschlüsselt dem anfragenden RADIUS-Server mitteilen. Der RADIUS-Server würde die vom SecOVID-Server erhaltene Antwort mit dem shared secret entschlüsseln und sich dann gemäß seiner Konfiguration verhalten.

- Konfiguration Ihres SecOVID-Servers (aus der Sicht des SecOVID-Servers ist der RADIUS-Server ein gewöhnlicher RADIUS-Client; siehe auch in den Abschnitten 3.1.1 und 3.1.2):
  - Registrieren Sie Ihren RADIUS-Server mit IP-Adresse und shared secret in `/etc/SecOvid/clients`.
  - Legen Sie die entsprechenden SecOVID-Benutzer mit dem Administrationstool in der SecOVID-Datenbank an.
  - Erzeugen Sie die entsprechenden Einmalpasswortgeneratoren auf den Chipkarten und geben Sie diese an die Endbenutzer aus.

  In der Grafik aus Abschnitt 1 sind die Kommunikationsflüsse zwischen den beteiligten Rechnern (Rechtecke) und Programmen (weiche Formen) schematisch dargestellt. In unserem Beispiel wäre die Firewall der NAS (Network Access Server). Die Kommunikation kann je nach Konfiguration der Firewall entweder direkt von der Firewall zum SecOVID-Server (durchgezogene Linie) oder von der Firewall über einen RADIUS-Server zum SecOVID-Server (gestrichelte Linie) verlaufen.

3.2.2 Konfiguration eines RADIUS-Clients*

Hinsichtlich der Konfiguration eines RADIUS-Clients gelten sinngemäß die Erläuterungen aus Abschnitt 3.2.4. Wenn ein RADIUS-Client Passwortanfragen an den SecOVID-Server weiterleiten soll, so müssen Sie

- ggf. als Passwortmethode an geeigneter Stelle „RADIUS“ auswählen
- die IP-Adresse des SecOVID-Servers als RADIUS-Authentifikationsserver eintragen
- ein shared secret für den SecOVID-Server eintragen (und dieses ebenso mit der IP-Adresse des RADIUS-Clients im SecOVID-Server eintragen)

Die letzten beiden Schritte werden für RADIUS-Clients auf UNIX-Systemen häufig durch Editieren der Datei `/etc/raddb/clients` vorgenommen.

3.2.3 Konfiguration eines TACACS+-Clients*

Hinsichtlich der Konfiguration eines TACACS+-Clients gelten sinngemäß die Erläuterungen aus Abschnitt 3.2.4. Wenn ein TACACS+-Client Passwortanfragen an den SecOVID-Server weiterleiten soll, so müssen Sie

- ggf. als Passwortmethode an geeigneter Stelle „TACACS+“ auswählen
- die IP-Adresse des SecOVID-Servers als TACACS+-Authentifikationsserver eintragen
- ein shared secret für den SecOVID-Server eintragen (und dieses ebenso mit der IP-Adresse des TACACS+-Clients im SecOVID-Server eintragen)

3.2.4 Konfiguration einer Firewall*

Ihre Firewall wird über die Möglichkeit verfügen, für jeden Benutzer einzustellen, welcher Passwortmechanismus zur Überprüfung dessen Passwörter von der Firewall verwendet werden soll.
Wenn die Passwörter eines Benutzers mit dem SecOVID-Einmalpasswortmechanismus überprüft werden sollen, wählen Sie für diesen Benutzer als Passwortmechanismus „RADIUS“ oder „TACACS+“ aus und tragen Sie als IP-Adresse für den RADIUS- oder TACACS+-Server die reale Adresse des Servers ein, der die entsprechenden Passwörter entgegennehmen und eine Antwort über die Gültigkeit der Passwörter zurücksenden soll. Dies kann direkt der SecOVID-Server sein oder aber ein RADIUS-Server, der Passwortanfragen seinerseits nach geeigneter Konfiguration zwecks Überprüfung an den SecOVID-Server weiterleitet. Darüber hinaus haben Sie in der Firewall (sowie im RADIUS- bzw. TACACS+- bzw. SecOVID-Server - je nachdem, mit wem die Firewall direkt kommuniziert) ein shared secret einzutragen.
Stellen Sie sicher, dass die RADIUS-konforme Kommunikation zwischen Firewall und SecOVID-Server nicht durch eine weitere Firewall behindert wird. Das bedeutet, die ggf. in der Mitte vorhandene Firewall muss all UDP-Pakete durchlassen, die von der zu stärkenden Firewall zum SecOVID-Server auf dessen RADIUS-Port (in der Regel Port 1812, siehe den Eintrag `radius` in der Datei `/etc/services` des SecOVID-Servers) gesendet werden. Zudem muss die mittlere Firewall alle UDP-Pakete durchlassen, die vom SecOVID-Server an die zu stärkende Firewall (undefinierter Port) zurückgesendet werden. Entsprechendes gilt für TCP-Pakete auf den TACACS+-Port (in der Regel Port 49) des SecOVID-Servers, sofern Sie TACACS+-Clients verwenden.

3.2.5 Konfiguration: Schutz der Einwahl über Windows 2000/XP/2003 RAS*

Wollen Sie Benutzern die Möglichkeit bieten, sich Ihrem Windows 2000/XP/2003 System einzuwählen, wobei die Authentifikation über SecOVID-Einmalpasswörter erfolgen soll, so können Sie dies einfach mit dem Routing and Remote Access Service (RRAS) erreichen. Hinweise zur Installation finden Sie in Abschnitt 2.3.5. Ein Benutzer, der sich von einem Client-Rechner einwählt, übergibt dann beim Einwählvorgang Benutzername und Einmalpasswort (statt Benutzername und Windows-Passwort) und wird darüber authentifiziert.
Wir beschreiben nun die Konfiguration des RRAS-Moduls auf dem Windows 2000 Server. Die Konfiguration des RRAS-Moduls unter Windows XP/2003 verläuft analog.
Die Konfiguration erfolgt zunächst so, als ob sich die Benutzer mit ihrem Windows Benutzerpasswort einwählen sollten. Sie können anschließend die notwendigen Einstellungen vornehmen, damit Benutzername und Passwort der Benutzer über die RADIUS-Schnittstelle an den SecOVID-Server weitergeleitet werden.
Konkret wählen Sie zur Konfiguration des RRAS-Moduls auf Ihrem Windows Server *Netzwerkumgebung* (rechte Maustaste) und anschließend *Eigenschaften* und *Dienste* aus. Markieren Sie *Routing and Remote Access Service* und klicken Sie auf *Eigenschaften*. Führen Sie hier die folgenden Schritte aus:

- Klicken Sie auf *Add* und registrieren Sie die gewünschten Modem-RAS-Verbindungen.
- Unter *Configure* markieren Sie lediglich *Receive calls as a RAS server.*
- Unter *Network* stellen Sie die Server Settings ein:
  - Markieren Sie unter *Allow remote clients running* alle Netzwerkprotokolle, die Sie für Clients unterstützen möchten, z.B. *TCP/IP* und *IPX*. Nehmen Sie zu jedem markierten Netzwerkprotokoll

unter dem Button *Configure* die notwendigen Netzeinstellungen vor und definieren Sie jeweils die für Clients zugelassenen Zielrechner zur Remote-Einwahl.

– Wählen Sie unter *Authentication and encryption settings* den Eintrag *Require encrypted authentication* aus.

– Unter *Authentication Provider* wählen Sie *RADIUS* (und nicht wie gewohnt *Windows*) aus. Mit diesem Eintrag weichen Sie in Ihrer Konfiguration zum ersten Mal vom Standardverfahren ab. Sie legen hier fest, dass anfragende Clients nicht durch den Windows Server, sondern durch einen RADIUS-Server authentifiziert werden sollen. Wählen Sie *configure* und dann *add* und tragen unter *Server Name* als RADIUS-Server den Namen oder die IP-Adresse Ihres SecOVID-Servers ein. Zudem tragen Sie unter *Secret* ein shared secret ein, das zur Verschlüsselung der RADIUS-konformen Kommunikation mit dem SecOVID-Server verwendet wird. Versäumen Sie nicht, den Windows Server mit seiner IP-Adresse und dem gleichen shared secret dem SecOVID-Server bekanntzumachen (durch Editieren der Datei `/etc/SecOvid/clients` auf dem SecOVID-Server). Zudem tragen Sie in *Timeout* ein, nach wie vielen Sekunden vergeblichen Wartens auf eine Antwort des SecOVID-Servers die Verbindung zum anfragenden RAS-Client abgebrochen werden soll. Akzeptieren oder editieren Sie den vorgeschlagenen Default-Wert unter *Initial Score*. Markieren Sie den Eintrag *Enable Authentication*. Tragen Sie schließlich unter *Port* die Portnummer ein, unter der Ihr SecOVID-Server RADIUS-Anfragen entgegennimmt (standardmäßig *1812*).

Installation und Konfiguration der RAS-Clients

Zur Installation und Konfiguration der üblichen RAS-Client-Software und des Modems bei den Endanwendern konsultieren Sie die Dokumentation von Microsoft bzw. die der Modems. Beachten Sie, dass Sie die Clients wie üblich zu installieren und zu konfigurieren haben, da die Benutzung des SecOVID-Servers für die Clientsoftware transparent ist, d.h. die Clientsoftware weiß nichts darüber, dass ihre Anfragen vom Windows Server zum SecOVID-Server weitergeleitet werden.

3.2.6 Konfiguration des Apache-Webservers*

3.2.6.1 Einleitende Erläuterungen

Falls Sie den SecOVID-Einmalpasswortmechanismus verwenden wollen, um Inhalte Ihres Apache Webservers oder Microsoft IIS nur den bei Ihnen registrierten SecOVID-Benutzern verfügbar zu machen, so haben Sie für den Webserver ein passendes Authentifikationsmodul zu installieren, damit der Webserver die Kommunikation mit dem SecOVID-Server lernt (RADIUS). Dieser Client bietet verschiedene Konfigurationsmöglichkeiten sowohl für unverschlüsselte als auch für mit SSL verschlüsselte Verbindungen. Verfügt der Benutzer über ein Benutzerzertifikat, so kann er wahlweise über sein Zertifikat oder ein SecOVID-Einmalpasswort authentifiziert werden. Inhalte des Webservers können beispielsweise ausschließlich mit dem SecOVID-Einmalpasswortmechanismus oder ausschließlich durch Verwendung der Benutzerzertifikate abgesichert werden. Zudem ist die Kombination beider Authentifikationsmethoden möglich. So kann der Benutzer nach geeigneter Konfiguration des Webservers z.B. von seinem heimischen PC auf abgesicherte Inhalte ihres Webservers zugreifen, sofern er sein gültiges Benutzerzertifikat vorweisen kann. Ist der Benutzer auf Dienstreise und verfügt nicht über sein Zertifikat, so kann die Authentifikation mittels SecOVID-Einmalpasswort durchgeführt werden. Darüber hinaus bestehen weitere Möglichkeiten zur Konfiguration: So muss beispielsweise nach der Authentifikation bei einer mit SSL gesicherten Verbindung zur Einsicht weiterer Dokumente im abgesicherten Bereich des Webservers kein weiteres Einmalpasswort vom Benutzer vorgelegt werden. Bei unverschlüsselter Verbindung kann die Zeitspanne der Gültigkeit eines Einmalpasswortes festgelegt werden. Nach Ablauf dieser Zeitspanne muss der Benutzer erneut ein gültiges Einmalpasswort vorlegen.

Bemerkung zur Verwendung von Benutzerzertifikaten:
Wenn Sie die oben beschriebene Kombination der Authentifikation mittels standardkonformem (X.509) Benutzerzertifikat und Einmalpasswort nutzen wollen, benötigen Sie zur Erstellung von Benutzerzertifikaten eine Trustcentersoftware, oder Sie müssen Benutzerzertifikate bei einem entsprechenden Dienstleister einkaufen. Sie

können hier beispielsweise die in Windows 2000 integrierte Trustcenter-Software (Windows 2000 CA) verwenden. Bei der Authentifikation mittels Benutzerzertifikat muss der Benutzer implizit Gebrauch von seinem privaten Schlüssel machen. Dieser muss unbedingt geheim bleiben. Normalerweise wird der geheime Schlüssel des Benutzers auf dessen Festplatte gespeichert, was ein großes Sicherheitsrisiko darstellt. Bei Verwendung des Produktes **KOBIL Smart Key** kann der geheime Schlüssel des Benutzers hochsicher auf der SecOVID-Chipkarte des Benutzers gespeichert werden. Aus Sicherheitsgründen empfehlen wir dringend die letztgenannte Variante. **KOBIL Smart Key** leistet insbesondere die Anbindung des Browsers des Benutzers an dessen Chipkarte. Beachten Sie, dass zur Verwendung von Benutzerzertifikaten in jedem Fall Soft- oder Hardware auf dem Computer des Benutzers installiert werden muss. Dieses raubt dem Benutzer die totale Mobilität, d.h. die Möglichkeit, von einem beliebigen Rechner aus ohne jegliche Installation auf geschützte Webseiten zuzugreifen. SecOVID gewährleistet hingegen die genannte totale Mobilität des Benutzers.

3.2.6.2 Schutz von Webseiten durch Einmalpasswörter auf dem Apache-Webserver

Wir wollen im folgenden annehmen, Ihr Apache-Webserver (Version 1.3.26) sei im Verzeichnis `/usr/local/apache-1.3.26` installiert. Nach korrekter Installation von SecOVID-Server, WWW-Server und Authentifikationsmodul können Sie den Zugriff auf ganze Verzeichnisse schützen. Die Webinhalte befinden sich im Verzeichnis `/usr/local/apache-1.3.26/htdocs/`. Dieses Verzeichnis bezeichnen wir im folgenden als ***DocumentRoot***. Soll der Zugriff auf Dateien, die auf dem WWW-Server liegen, nur nach vorheriger Verifikation eines Einmalpasswortes erfolgen, so gehen Sie folgendermaßen vor:

1. Legen Sie ein Verzeichnis unterhalb ihres ***DocumentRoot***–Verzeichnisses an, z.B. ***DocumentRoot***`/meyer`. Nehmen wir an, Sie möchten dieses Verzeichnis nur Herrn Meyer mit dem SecOVID-Account `meyer` zugänglich machen, und niemand sonst soll Zugriff auf dieses Verzeichnis erhalten. Kopieren Sie alle Webinhalte, die durch die Einmalpasswörter von Herrn Meyer geschützt werden sollen, in das Verzeichnis ***DocumentRoot***`/meyer`

2. Jetzt wird der Zugriff auf das entsprechende Verzeichnis geschützt: Legen Sie im zu schützenden Verzeichnis eine Datei mit dem Namen `.htaccess` an. Aus dieser Datei liest der Apache-Webserver verzeichnisspezifische Konfigurationen, sofern dies in der Datei `/usr/local/apache-1.3.26/conf/httpd.conf` explizit gestattet ist. Dies ist in der von KOBIL gelieferten Version des Apache-Webservers der Fall. (Der Konfigurationsparameter `AllowOverride All` ist gesetzt.)
   Dieses Argument kann in verschiedenen Abschnitten in der `/usr/local/apache-1.3.26/conf/httpd.conf` gesetzt werden.
   Abschnitte werden in der `/usr/local/apache-1.3.26/conf/httpd.conf` mit Markern gesetzt:
   Ein Abschnitt fängt z.B. an mit `<Directory "/usr/local/apache-1.3.26/htdocs">` und endet mit `</Directory>`. Alle Anweisungen, die zwischen den Markern liegen, gelten nur in diesem Bereich. Der von uns gelieferte Apache Server hat in diesem Bereich folgende Konfiguration:

   ```
   <Directory "/usr/local/apache-1.3.26/htdocs">
              Options Indexes FollowSymLinks MultiViews
              AllowOverride All
              Order allow,deny
              Allow from all
   </Directory>
   ```

   Die Grundkonfiguration des Webservers wird in der Datei `/usr/local/apache-1.3.26/conf/httpd.conf` vorgenommen. Wenn die Option `AllowOveride All` gesetzt ist, wird zum Schutz von Verzeichnissen unterhalb von `/usr/local/apache-1.3.26/htdocs/` die Konfiguration der lokalen Datei `.htaccess` verwendet (Diese Datei `.htaccess` muss also im zu schützenden Verzeichnis angelegt sein). Bezüglich weiterer Konfigurationsmöglichkeiten verweisen wir auf die Dokumentation des Apache-Servers.

3. Jetzt beschreiben wir den Inhalt der Datei mit dem Namen `.htaccess` im zu schützenden Verzeichnis. **Beachten Sie, dass Sie nach Installation des Webservers bereits über ein vollständiges, sofort funktionierendes Beispiel verfügen, wie man den Zugriff auf Webseiten dem SecOVID-Schutz unterstellt, wenn Apache-Server und SecOVID-Server auf dem gleichen Rechner laufen.** Insbesondere ist bereits eine Datei `.htaccess` enthalten, die Sie an die richtige Stelle kopieren und ggf. nachträglich editieren können. Diese Datei muss folgende Einträge enthalten:

```
AuthType              Basic
AuthName              SecOvid
AuthSecOvid           my_secovid_server.my_domain.de
SecOvidPort           1812
RequireSessionKey     on
SecOvidTimeout        60
require user meyer
```

Diese Einträge haben folgende Bedeutung:

- `AuthType Basic` ist ein Apache-interner Eintrag.
- `AuthName` definiert den Titel des Authentifizierungsfensters, also des Fensters, in dem der Endanwender Benutzernamen und Einmalpasswort einzugeben hat, um Zugriff auf eine Webseite zu erhalten.
- `AuthSecOvid` definiert die IP-Adresse oder den Namen des SecOVID-Servers, der die Authentifikation durchführen soll.
- `SecOvidPort` definiert die Portnummer des SecOVID-Servers, an welche Authentifizierungsdaten geschickt werden. Geben Sie hier die Nummer des Ports ein, an dem der SecOVID-Server RADIUS-Anfragen erwartet. Maßgeblich ist der Eintrag `radius` in der Datei `/etc/services` des SecOVID-Servers.
- `RequireSessionKey`: Die größtmögliche Sicherheit beim Zugriff auf Webseiten wird erreicht, wenn der Endanwender (z.B. durch entsprechende Konfiguration seines Browsers) das Abspeichern von Cookies auf seinem Rechner erlaubt, denn Cookies werden in unserem Kontext zum Generieren von Session Keys verwendet, die eine zuverlässigere Identifikation des Rechners des Endanwenders erlauben. Der Eintrag `RequireSessionKey` legt fest, was passieren soll, wenn der Endanwender auf eine geschützte Seite zugreifen will, aber das Abspeichern von Cookies verboten hat. Folgt hinter dem genannten Eintrag `on`, so ist ein Session Key für die Übertragung der Daten zwingend erforderlich, und der Apache-Server gibt die Webseite nicht frei. Ist hingegen der Wert `off` gesetzt, so gibt der Apache-Server die Webseite bei Vorlage eines gültigen Einmalpasswortes für die Anzahl der Sekunden frei, die hinter dem Eintrag `SecOvidTimeout` eingetragen ist, auch wenn der Endbenutzer das Abspeichern von Cookies verboten hat. Aus Sicherheitsgründen empfehlen wir, den Wert `on` zu setzen und den Endanwendern nahezulegen, das Abspeichern von Cookies durch den Browser zuzulassen.

  Warnung: Bei diesem Feature kann es zu folgenden Problemen kommen. Der Apache-Webserver führt bei einem Link der Form `<a href="/safedoc/meyer/privat/">` mehrere Zugriffe aus, nämlich jeweils einen Zugriff in die Verzeichnisse `safedoc`, `meyer` und `privat`. Anschießend wird nach einer Datei `index.html` im Verzeichnis `/safedoc/meyer/privat/` gesucht. Insgesamt werden in unserem Beispiel demnach drei Zugriffe durchgeführt. Befindet sich die Datei `.htaccess` im Verzeichnis `safedoc`, wird das Cookie bei dem Zugriff auf `/safedoc/meyer/privat/` nicht rechtzeitig übertragen. Daher wird dem Benutzer kein Zugriff gestattet.

  Lösung des Problems:
  Ersetzen Sie den Link, bei dem das Problem auftritt, z.B. `<a href="/safedoc/meyer/privat/">` durch `<a href="/safedoc/meyer/privat/index.html">`.

  Damit beseitigen Sie dieses Problem. Zudem beschleunigen Sie den Zugriff auf das betreffende Verzeichnis Ihres Webservers.
- `SecOvidTimeout` definiert die Zeit (in Sekunden), für die der Endanwender bei Vorlage eines gültigen Einmalpasswortes Zugriff auf SecOVID-geschützte Seiten erhält, die nicht per SSL übertragen werden.

Nach Ablauf dieser Zeitspanne verliert der Endanwender sein Zugriffsrecht auf die geschützten Seiten und wird zur Eingabe eines neuen Einmalpasswortes aufgefordert.

- `require user meyer` gibt an, dass der Apache-Server eine Seite für den User `meyer` freigeben soll, der durch den SecOVID-Server authentifiziert werden konnte (mit der Ausnahme, dass `RequireSessionKey on` gesetzt ist und der User das Abspeichern von Cookies verbietet).

  Ihnen stehen weitere Konfigurationsmöglichkeiten offen:
  `require valid-user` gibt an, dass der Apache-Server eine Seite für einen beliebigen User freigeben soll, der durch den SecOVID-Server authentifiziert werden konnte (mit der Ausnahme, dass `RequireSessionKey on` gesetzt ist und der User das Abspeichern von Cookies verbietet).
  `require user meyer neis binisik` würde den Benutzern `meyer, neis` und `binisik` den Zugriff auf das jeweilige Verzeichnis gestatten, sofern einer dieser Benutzer ein gültiges Einmalpasswort vorlegt. Existieren keine weiteren Einträge, die mit `require` beginnen, so werden keine anderen als die hier genannten Benutzer Einlass erhalten, auch wenn sie durch den SecOVID-Server authentifiziert werden konnten.
  `require group groupname1 groupname2` gibt an, dass der Apache-Server eine angeforderte Seite für einen User freigeben soll, der mindestens einer der angegebenen Apache-internen Gruppen angehört und durch den SecOVID-Server authentifiziert werden konnte. Die Liste der Gruppennamen kann beliebig lange sein. Existieren keine weiteren Einträge, die mit `require` beginnen, so werden keine anderen als die hier genannten Benutzer Einlass erhalten, auch wenn sie durch den SecOVID-Server authentifiziert werden konnten.

4. Stellen Sie ggf. durch einen entsprechenden Eintrag in der Datei `/etc/hosts` sicher, dass der SecOVID-Server unter obigem Namen bekannt ist:

   ```
   192.168.88.164     my_secovid_server.my_domain.de     my_secovid_server
   ```

5. Editieren Sie die bereits vorhandene Datei
   `/usr/local/apache-1.3.26/conf/httpd.conf` , so
   dass die folgenden Einträge an den entsprechenden Stellen vorhanden sind:

   ```
   [...]
   LoadModule secovid_auth_module  libexec/mod_auth_secovid.so
   AddModule mod_auth_secovid.c
   <Directory "/usr/local/apache-1.3.26/htdocs">
   [...]
   AllowOverride All
   [...]
   </Directory>
   ```

6. Tragen Sie in der Datei `/etc/SecOvid/clients` ein shared secret für den SecOVID-Server ein. Falls die Datei noch nicht existiert, legen Sie eine entsprechende Datei an. Tragen Sie das gleiche shared secret auf dem SecOVID-Server in dessen Datei `/etc/SecOvid/clients` ein. Informationen zur Syntax der Datei finden Sie in Abschnitt 3.1.1.

7. Starten Sie den Apache-Server mit
   `/usr/local/apache-1.3.26/bin/apachectl startssl`

Testen der Funktionalität des WWW-Zugriffsschutzes

Angenommen, `www.mycompany.de` ist der Name Ihres gerade konfigurierten Apache-Servers, dann starten Sie mittels eines Browsers von einem anderen Rechner aus eine Anfrage auf den WWW-Server:

https://www.mycompany.de
Ihnen wird nun angeboten, sich das Zertifikat von Ihrem WWW-Server zu laden. Folgen Sie hier den Anweisungen auf dem Monitor und akzeptieren Sie das Zertifikat testweise. Sie können sich zu einem späteren Zeitpunkt ein neues Zertifikat für Ihren WWW-Server organisieren. Mit dem bereits vorinstallierten Zertifikat können Sie vorab die Funktionsweise testen.

Bemerkung: Für weitergehende Konfigurationshinweise konsultieren Sie die Dokumentation Ihres Apache-Servers, siehe `www.apache.org`

Schutz von Webseiten durch Kombination von Zertifikaten und Einmalpasswörtern

Falls sie den Schutz ihrer Webinhalte durch kombinierte Benutzung von Zertifikats- und Einmalpasswort-basierter Authentifikation gewährleisten wollen, so gehen Sie wie folgt vor:

1. Legen Sie im zu schützenden Verzeichnis eine Datei `.htaccess` an, die folgende Zeilen enthält:

```
SSLVerifyClient optional
SSLOptions  +FakeBasicAuth +StrictRequire
AuthType               Basic
AuthName               SecOvid
AuthSecOvid            my_secovid_server.my_domain.de
SecOvidPort            1812
RequireSessionKey      on
SecOvidTimeout         60
SecOvidSSLClientsFile  /etc/SecOvid/sslclientsfile
SecOvidAllowSSLClients meyer neis binisik tak.*
require valid-user
```

2. Legen Sie das Verzeichnis `/etc/SecOvid/` und eine Datei `/etc/SecOvid/sslclientsfile` an.

3. Registrieren Sie nun die Benutzer, die über Benutzerzertifikate verfügen, in der Datei `/etc/SecOvid/sslclientsfile`.
   Die Einträge haben folgende Gestalt:

```
meyer:/C=DE/ST=fasdfa/L=opiopi/O=452354/OU=98asdf/CN=fadf:1:1
binisik:/C=DE/ST=Rheinland-Pfalz/L=Worms/O=Kobil/OU=www/CN=binisik:1:1
emre:/C=DE/ST=Rheinland-Pfalz/L=Worms/O=Kobil/OU=www/CN=binisik:1:1
tak:/C=DE/ST=Rheinland-Pfalz/L=Worms/O=Kobil/OU=development/CN=Markus Tak:1:0
tak-admin:/C=DE/ST=Rheinland-Pfalz/L=Worms/O=www/OU=administration/CN=Markus Tak:0:1
```

   Die Syntax und Bedeutung der Einträge lautet wie folgt:

   - Jede Zeile beschreibt einen Benutzer.
   - Jede Zeile enthält 4 Felder, die mit einem Doppelpunkt ':' getrennt sind (ähnlich einer Passwort-Datei in UNIX).
   - Das 1. Feld ist der Loginname des Benutzers. Unter diesem Namen kann sich der Benutzer beim SecOVID-Server authentifizieren.
   - Das 2. Feld ist der DN (Distinguished Name) des Benutzerzertifikats. Dieser dient zur eindeutigen Unterscheidung der Benutzer durch ihre Zertifikate.
   - Das 3. Feld ist ein Schalter mit den Werten 0 oder 1. Ist der Wert 0 gesetzt, so ist der Zugang auf WWW-Seiten für diesen Benutzer generell untersagt (der Benutzer ist auf dem Webserver gesperrt). Ist der Wert 1 gesetzt, so ist der Zugang prinzipiell erlaubt (Der Benutzer hat sich natürlich zu authentifizieren).

- Das 4. Feld ist ebenfalls ein Schalter mit den Werten 0 oder 1. Steht er auf 1, ist es dem Benutzer gestattet, Einmalpasswörter statt Zertifikate zur Authentifizierung zu verwenden. Steht der Eintrag in diesem Feld auf 0, so kann der Benutzer sich nicht mit Einmalpasswörtern authentifizieren.

Die Variable `SecOvidAllowSSLClients` enthält die Liste aller Benutzer, die auf das Verzeichnis zugreifen können. Hier können reguläre Ausdrücke verwendet werden. Im Beispiel gestattet der Eintrag tak.* den Benutzern in den Zeilen:

```
tak:/C=DE/ST=Rheinland-Pfalz/L=Worms/O=Kobil/OU=development/CN=Markus Tak:1:0
tak-admin:/C=DE/ST=Rheinland-Pfalz/L=Worms/O=www/OU=administration/CN=Markus Tak:0:1
```

den Zugang. Bei Nicht-Verwendung dieser Variablen dürfen alle Benutzer aus der mittels `SecOvidSSLClientsFile` spezifizierten Datei (hier `/etc/SecOvid/sslclientsfile`) auf das Verzeichnis zugreifen.

3.2.7 UNIX-Schutz: Benutzung der Arbeitsplatzrechner*

Nach den in Abschnitt 2.2.8 beschriebenen Eingriffen in einen UNIX-Host müssen Endanwender, die sich gegenüber dem UNIX-Host authentifizieren wollen (z.B. beim lokalen Login oder beim telnet), Username und Passwörter vorlegen. Die Art der Passwörter, die zum erfolgreichen Einloggen vom Endanwender angegeben werden müssen, hängt von der Konfiguration des UNIX-Hostes ab (siehe Abschnitt 2.3.9). Geeignete Informationen zeigen dem Endanwender an, welche Art von Passwort einzugeben ist. Beispielsweise wird der Endanwender ggf. mit **Secovid OTP:** zur Eingabe eines SecOVID-Einmalpasswortes (one-time password) aufgefordert.

Bitte beachten Sie erneut, dass ein Einloggen nur möglich ist, wenn die Dateien `ovid_config` und `ovid_comm.pub` an den erwarteten Stellen vorhanden sind. Beachten Sie insbesondere, dass diese Dateien für den jeweiligen Benutzer lesbar sein müssen, damit er beispielsweise xscreensaver wieder beenden kann (siehe Abschnitt 2.3.9).

3.2.8 Konfiguration eines RSA-Clients*

Hier gelten sinngemäß die Erläuterungen aus Abschnitt 2.1.7. Wenn ein RSA-Client Passwortanfragen an den SecOVID-Server weiterleiten soll, so müssen Sie

- die IP-Adresse des SecOVID-Servers und die Portnummer, auf der der SecOVID-Server RSA-Anfragen entgegennimmt (vgl. Eintrag secovid in `/etc/services` auf dem SecOVID-Server) eintragen. In der Regel geschieht dies durch Editieren der Datei `ovid_config`. IP-Adresse und Portnummer sind dann durch Doppelpunkt zu trennen.
- den Public Key des SecOVID-Servers eintragen. In der Regel haben Sie hierzu die Datei `/etc/SecOvid/ovidcomm.pub` des SecOVID-Servers an die richtige Stelle zu kopieren.

Ein Beispiel für die durchzuführenden Schritte finden Sie in Abschnitt 2.1.7. Dort wird das einfache RSA-Clientprogramm `ovid_ping` konfiguriert.

3.3 Das GetPWD Programm

3.3.1 Linux und Solaris

Wenn der Benutzer das Programm `getpwd` startet, erscheint ein Fenster auf dem Bildschirm (in der kurzen Wartezeit wird überprüft, was für ein Chipkartenterminal angeschlossen ist und ob eine Chipkarte in das Terminal eingelegt ist). Besitzt das angeschlossene Terminal eine Tastatur, so muss der Benutzer, wenn er ein Passwort erzeugen will, den Button „OTP“ drücken und die PIN am Terminal eingegeben. Wird ein Terminal ohne Tastatur verwendet, so startet das Dialogfenster mit einem Eingabe-Fenster, in den man die PIN über die Tastatur des Computers eingeben kann. Bei jedem Drücken des OTP-Buttons wird nun ein SecOVID-Einmalpasswort erzeugt. Sobald das Einmalpasswort berechnet ist, kann dieses mit Hilfe der Maus an der gewünschten Stelle eingefügt werden. Unter UNIX-Applikationen, die „Drag&Drop“ unterstützen, geschieht dies, indem man in das Passwort-Fenster klickt, in dem das Passwort durch Sternchen symbolisiert wird und das Passwort an die gewünschte Stelle zieht. Für eine UNIX-Applikation ohne „Drag&Drop“-Unterstützung (z.B. „telnet“ in einem „xterm“) wird das Passwort zudem als Standard-Einfüge-Text exportiert, d.h. es lässt sich durch einen Klick auf die mittlere Maustaste einfügen.
Will man das Passwort im Klartext angezeigt haben, so kann man diese Einstellung nach Drücken des Optionen-Button vornehmen. Beim Erzeugen des nächsten Passworts, wird das SecOVID-Passwort nun im Klartext angezeigt.

3.3.2 Windows

Wenn der Benutzer das Programm `getpwd` startet, so erscheint kein Dialog-Fenster auf dem Bildschirm, sondern das Programm wird gestartet und als Icon in der Windows-eigenen Taskleiste dargestellt. Durch einen Doppelklick mit der linken Maustaste auf das Icon erscheint das Eingabe-Fenster, das sich wie in Abschnitt

3.3.1 beschrieben verhält, wobei unter Windows neben der „Drag&Drop“-Option die Möglichkeit besteht, das Passwort mit „Strg + V“ an die gewünschte Stelle zu kopieren.
Drückt man den Button „Ausblenden“, so verschwindet das Fenster wieder und wird als Icon in der Taskleiste abgelegt.
Bei der Verwendung eines Terminals mit Tastatur besteht außerdem die Möglichkeit, ein Passwort zu generieren, indem man mit der rechten Maustaste auf das Icon in der Taskleiste drückt und im erscheinenden Menü die Option „OTP generieren“ drückt. Ohne dass das Dialog-Fenster auf dem Bildschirm erscheint, kann man die PIN am Terminal eingeben und das erzeugte SecOVID-Einmalpasswort über „Strg + V“ an die gewünschte Stelle kopieren.

Kapitel 4

Deinstallation

Warnung:
Wenn Sie die SecOVID-Software dauerhaft deinstallieren wollen, beachten Sie, dass Sie zunächst alle eventuell installierten Module zum Schutz von Arbeitsplatzrechnern deinstallieren sollten, ansonsten können Sie sich aus Ihrem System aussperren. Auch wenn Sie den Server nur vorübergehend deaktivieren wollen, sollten Sie überlegen, welche Client-Software nun (vorübergehend) nicht funktionieren wird und ob ggf. Maßnahmen wie Umkonfigurationen notwendig sind, um diesen Ausfall zu überbrücken.

Zur Deinstallation der aller auf Windows-Rechnern installierten Module nutzen Sie die betriebssystemseitige Unterstützung zur Deinstallation von Software (*Start* > *Einstellungen* > *Systemsteuerung* > *Software* > *Deinstallation* > *Entfernen*).
Zur Deinstallation des Unix-Schutzes (falls Sie ihn installiert haben) müssen Sie die PAM-Konfigurationsdatei dahingehend editieren, dass das Modul `pam_ovid_auth.so` nicht mehr (und gegebenenfalls ein anderes PAM Modul z.B. `pam_unix.so` wieder) benutzt wird (vgl. Abschnitt 2.3.9). Verifizieren Sie dann, dass Sie sich nun ohne Einmalpasswort anmelden können. Danach können Sie auf den Clients die Dateien
`/usr/lib/security/pam_ovid_auth.so`, `/etc/SecOvid/ovidcomm.pub` und `/etc/SecOvid/ovid_config` entfernen.
Wenn die Möglichkeit des Remote-Zugriffs für Ihr internes Netzwerk stark genutzt wird, sollten Sie auch Router oder Firewalls (oder sonstige relevante TACACS+- oder RADIUS-Clients) zunächst umkonfigurieren.
Evtl. vorhandene Webserver und sonstige Clienten können Sie wahrscheinlich auch nach der Deinstallation des Servers bei Auftreten von Problemen umkonfigurieren, idealerweise konfigurieren Sie aber auch diese so um, dass sie unabhängig vom SecOVID-Server funktionieren, bevor Sie sich daran machen, den Server zu deinstallieren.

Nun können Sie den SecOVID-Server wie in Abschnitt 3.1.1 beschrieben stoppen. Wollen Sie ihn nur vorübergehend deaktivieren, so sorgen Sie nun dafür, dass beim Booten des Rechners nicht mehr versucht wird, den SecOVID-Dämon zu starten, beispielsweise durch Entfernen der betreffenden Datei aus den Verzeichnissen mit den automatisch beim Booten ausgeführten Skripten, auf Linux-System etwa mittels des Kommandos

```
mv /etc/rc.d/rc2.d/S65secovid /etc/SecOvid/S65secovid.bak
```

und auf Solaris-System mittels

```
mv /etc/rc2.d/S65secovid /etc/SecOvid/S65secovid.bak
```

Zur Sicherung Ihrer Benutzer- und Karten- oder Tokendaten können Sie nun ein Backup der Datei `/etc/SecOvid/ovid_data` machen, wenn Sie die Option einer späteren Neu-Installation aufrecht erhalten wollen, können Sie auch ein Backup der sonstigen Konfigurationsdateien im `/etc/SecOvid`-Verzeichnis machen.
Zur Deinstallation des SecOVID-Serversystems (mitsamt dem Admintool) entfernen Sie ggf. die von SecOVID installierten Libraries aus den Verzeichnissen, die Sie bei der Installation des SecOVID-Servers und der SecOVID-Administrationstools angegeben hatten. Die Pfade zu den entsprechenden Verzeichnissen finden Sie in den Startupskripten `/etc/SecOvid/secovid.sh` und `/etc/SecOvid/SecOvid_admin.sh`. Abschließend löschen Sie nun die verbliebenen Dateien und Directories mittels

```
rm -rf /etc/SecOvid/
```

Nun können Sie auch auf den Rechnern, auf denen der SecOVID-Server oder Clientsoftware installiert war, folgende Einträge aus der Datei `/etc/services/` entfernen (bzw. auskommentieren):

```
tacacs              49/tcp
tacacs              49/udp
radius              1812/udp
secovid             1647/udp
secoviddb           1113/udp
```

Kapitel 5

Lizenzvereinbarung

Die gelieferte bzw. eingerichtete Software und die dazugehörige Dokumentation sind urheberrechtlich durch KOBIL Systems geschützt. Die Nutzung und das Kopieren beider Produkte darf nur in Einklang mit den Lizenzbedingungen (Internationale Nutzungsbedingungen) und unter Einbezug des „Copyright“ vorgenommen werden. Die Software und das mitgelieferte Handbuch darf Dritten nicht zur Verfügung gestellt werden. Darüber hinaus ist es nicht gestattet, die durch KOBIL Systems eingeführten Titel, Überschriften und Bezeichnungen weiter zu verwenden. Jede unbefugte Nutzung oder Reproduktion der Software oder der Dokumentation wird juristisch verfolgt. Die Informationen, die in der Software oder in dem Handbuch enthalten sind, erheben nicht den Anspruch auf Vollständigkeit.
©2003 by KOBIL Systems GmbH. Alle Rechte vorbehalten.
Hergestellt in Deutschland.
Internationale Nutzungsbedingungen der KOBIL Systems Lesen Sie vor dem Verwenden der Software diese Vereinbarung sorgfältig durch. KOBIL Systems erteilt Ihnen die Lizenz für diese Software nur, wenn Sie die Bedingungen dieser Vereinbarung akzeptiert haben. Durch die Verwendung der Software erklären Sie sich mit diesen Bedingungen einverstanden.
Die Software ist Eigentum der KOBIL Systems GmbH oder eines Lieferanten von KOBIL Systems; sie ist urheberrechtlich geschützt und wird lizenziert, nicht verkauft.
Unter „Software“ sind im Sinne dieser Vereinbarung die Originalsoftware und alle vollständigen oder teilweisen Kopien hiervon zu verstehen. Software besteht aus maschinenlesbaren Anweisungen, Daten, akustischem/ optischem Inhalt (z. B. Grafiken, Text, Aufnahmen oder Bilder) und zugehörigem lizenziertem Material.

5.1 Lizenz

Nutzung der Software
Sie erhalten von der KOBIL Systems ein nicht ausschließliches Recht zur Nutzung der Software. Sie sind berechtigt, 1) die Software im Rahmen der erworbenen Berechtigungen zu nutzen und 2) für den Verwendungsumfang erforderliche Kopien zu erstellen und zu installieren, sofern Sie den Copyrightvermerk und eventuelle andere Eigentumshinweise auf jeder Kopie oder teilweisen Kopie der Software anbringen. Bei Erwerb dieser Software als Upgrade endet die Berechtigung zur Nutzung der Software, auf dessen Basis das Upgrade erfolgte. Sie verpflichten sich, dass jeder Benutzer die Software nur bestimmungsgemäß verwendet und die Bedingungen dieser Vereinbarung beachtet. Sie sind nicht berechtigt, 1) die Software in anderer Weise als hierin beschrieben zu nutzen, zu kopieren, zu ändern oder weiterzugeben; 2) die Software zu disassemblieren oder zu decompilieren oder in anderer Weise zu übersetzen, sofern eine solche Übersetzung nicht durch ausdrückliche gesetzliche Regelung unabdingbar vorgesehen ist; 3) Unterlizenzen für die Software zu erteilen und sie zu vermieten oder zu verleasen.

5.2 Berechtigungsnachweis

Der Berechtigungsnachweis (Rechnung) für diese Software ist als Nachweis des Anspruchs auf Garantieleistungen von der KOBIL Systems oder eines dazu berechtigten Wiederverkäufers, zukünftige Preise für künftige Software-Upgrades (falls angekündigt) und mögliche Sonder- oder Werbeaktionen aufzubewahren. Ferner gilt er als Nachweis des Benutzers für die berechtigte Nutzung dieser KOBIL Systems Lizenzsoftware.

5.3 Gebühren und Abgaben

Je nach Anzahl der von der Software registrierten Endbenutzer wird die Berechnung festgelegt und im Berechtigungsnachweis nachgewiesen. Die Gebühren richten sich nach der maximal möglichen Anzahl an registrierten Endbenutzern. Wenn Sie diese maximale Anzahl erhöhen wollen, verständigen Sie KOBIL Systems oder deren Wiederverkäufer. Die Erweiterung ist gebührenpflichtig. Bereits fällige oder bezahlte Beträge werden gutgeschrieben.

5.4 Gewährleistung

KOBIL Systems garantiert, dass die Software bei Nutzung in der im Installationshandbuch angegebenen Betriebsumgebung ihrer Spezifikation entspricht. KOBIL Systems garantiert keinen ununterbrochenen oder fehlerfreien Betrieb der Software oder die Behebung aller Softwarefehler. Für die Ergebnisse aus der Nutzung der Software sind Sie selbst verantwortlich. Die Gewährleistungsdauer für die Software läuft sechs Monate nach Erwerb der Software ab. Während der Gewährleistungsdauer ist der Garantieservice für den unveränderten Teil der Software durch fehlerbezogenen Softwareservice kostenfrei. Softwareservice steht mindestens ein Jahr lang nach der allgemeinen Verfügbarkeit der Software zur Verfügung. Wenn die Software innerhalb eines Jahres nach Erwerb der Lizenz nicht wie zugesichert funktioniert und KOBIL Systems das Problem mit einer Korrektur, Einschränkung oder Umgehung nicht lösen kann, sind Sie berechtigt, die Software an KOBIL Systems oder den KOBIL Systems-Wiederverkäufer, von dem Sie die Software erworben haben, zurückzugeben. Sie erhalten den Kaufpreis zurück. Diese Gewährleistung ersetzt alle sonstigen geltenden Gewährleistungsbedingungen, seien Sie veröffentlicht oder stillschweigend gültig, einschließlich, aber nicht begrenzt auf die implizierte Gewährleistung für die Handelsüblichkeit und die Verwendungsfähigkeit für einen bestimmten Zweck. Mit diesen Gewährleistungsbedingungen werden ausdrücklich bestimmte Rechte erteilt, und außerdem können noch andere Rechte gelten, die von Rechtsordnung zu Rechtsordnung variieren. Einige Rechtsordnungen erlauben nicht den Ausschluss von stillschweigenden Gewährleistungsansprüchen oder deren Beschränkung, so dass der/ die obige Ausschluss/ Beschränkung nicht anwendbar ist. In diesem Fall sind solche Gewährleistungen auf die Dauer der Gewährleistungsperiode befristet. Nach Ablauf dieser Periode gibt es keine Gewährleistung mehr.

5.5 Haftungsbeschränkung

Unabhängig vom Grund eines Schadensersatzanspruches gegen die KOBIL Systems (einschließlich grundlegendem Vertragsbruch, Fahrlässigkeit, falscher Darstellung oder anderem Rechtsbruch) ist die Haftung der KOBIL Systems wie nachfolgend aufgeführt begrenzt auf 1) Schadenersatz für Personenschäden und Schäden an Immobilien und nicht immateriellen, beweglichen Sachen und 2) andere direkte Schäden bis zu einer Summe von EUR 12.500 (oder entsprechender Betrag in der Landeswährung) bzw. auf die Gebühren für die Software, die Gegenstand des Anspruchs ist. KOBIL Systems haftet nicht für spezielle mittelbare oder Folgeschäden oder andere geschäftliche Folgeschäden (einschließlich entgangenem Gewinn oder entgangenen Einsparungen), selbst wenn die KOBIL Systems oder Ihr Wiederverkäufer über die Möglichkeit solcher Schäden informiert wurde. KOBIL Systems haftet nicht für 1) den Verlust oder die Beschädigung von Daten, 2) irgendwelche Schadensersatzansprüche basierend auf Ansprüchen Dritter. Diese Haftungsbeschränkung gilt auch für Entwickler von Software, die an KOBIL Systems geliefert wird. Dies ist die Höchstgrenze, für die KOBIL Systems und ihre Lieferanten gemeinsam haftbar gemacht werden können.

5.6 Allgemein

Keine Bestimmung in dieser Vereinbarung betrifft Verbraucherschutzrechte, die gesetzlich unabdingbar sind. KOBIL Systems kann Ihre Lizenz beenden, wenn Sie die Bedingungen in dieser Vereinbarung nicht einhalten. In diesem Fall erlischt auch Ihre Berechtigung zur Nutzung der Software. Sie verpflichten sich, die geltenden Exportgesetze und -bestimmungen einzuhalten. Sie und KOBIL Systems verpflichten sich im Rahmen dieser Vereinbarung zu einer Verjährungsfrist von zwei Jahren nach Eintreten des Klagegrunds. Gesetzliche Fristen bleiben unberührt. Weder Sie noch KOBIL Systems sind verantwortlich für nicht erfüllte Verpflichtungen auf Grund höherer Gewalt.

Kapitel 6

Support

Falls Sie technischen Support benötigen, konsultieren Sie bitte zunächst die FAQ-Liste (frequently asked questions) unter `www.kobil.com`.

Hilft dies nicht weiter, wenden Sie sich an Ihren Händler, der Ihnen First Level Support gewährt.

Adresse des Herstellers:
KOBIL Systems GmbH
Pfortenring 11
D-67547 Worms
www.kobil.com

Wenn Sie Support in Anspruch nehmen, halten Sie bitte die folgenden Informationen verfügbar. Falls Sie per E-Mail Kontakt aufnehmen, senden Sie die folgenden Informationen unbedingt mit:

1. Die Nummer Ihres Support-Vertrages
2. Ihre SecOVID-Softwareversionsnummer. Sie finden diese Nummer auf Ihrer SecOVID-CD ROM.
3. Die ausführlichen Versionsinformationen über SecOVID-Server und SecOVID-Administrationstool. Sie erhalten diese Versionsinformationen unter Linux mit dem Kommando `ident` und unter Solaris mit dem Kommando `what`. Geben Sie Sie unter Linux also `ident /etc/SecOvid/secovid` und `ident /etc/SecOvid /admin/WxOvid` ein und notieren Sie die Ausgaben der Kommandos.

   Sollten auf ihrem Computer `ident` bzw. `what` nicht installiert sein, so können Sie, falls zumindest `strings` vorhanden ist, im allgemeinen die gleiche Information auch durch ein Kommando wie `strings /etc/SecOvid /secovid | grep \$Id:` erhalten.
4. Die Marke und das Modell Ihres Computers sowie die Version des Betriebssystems, unter dem das Problem auftritt. (Diese Informationen erhalten Sie z.B. mit dem UNIX-Kommando `uname -a`)